




Universitat Autònoma de Barcelona

**ADVERTIMENT.** L'accés als continguts d'aquesta tesi queda condicionat a l'acceptació de les condicions d'ús establertes per la següent llicència Creative Commons:  [http://cat.creativecommons.org/?page\\_id=184](http://cat.creativecommons.org/?page_id=184)

**ADVERTENCIA.** El acceso a los contenidos de esta tesis queda condicionado a la aceptación de las condiciones de uso establecidas por la siguiente licencia Creative Commons:  <http://es.creativecommons.org/blog/licencias/>

**WARNING.** The access to the contents of this doctoral thesis it is limited to the acceptance of the use conditions set by the following Creative Commons license:  <https://creativecommons.org/licenses/?lang=en>



**Universitat Autònoma  
de Barcelona**

Estimating Light Effects from a Single  
Image: Deep Architectures and  
Ground-Truth Generation

A dissertation submitted by **Hassan Ahmed Sial** at  
Universitat Autònoma de Barcelona to fulfil the de-  
gree of **Doctor of Philosophy**.

Bellaterra, June 30, 2021

Directors	<b>Dr. Maria Vanrell</b> <b>Dr. Ramon Baldrich</b> Centre de Visió per Computador
Thesis committee	<b>Dr. Marius Pedersen</b> Department of Computer Science The Norwegian University of Science and Technology  <b>Dr. Javier Vazquez Corral</b> Centre de Visió per Computador Universitat Autònoma de Barcelona  <b>Dr. Francesc Moreno-Noguer</b> Institut de Robòtica i Informàtica Industrial Universitat Politècnica de Catalunya
Supplement	<b>Dr. Joost van de Weijer</b> Centre de Visió per Computador Universitat Autònoma de Barcelona  <b>Dr. Olivier Penacchio</b> School of Psychology and Neuroscience University of St Andrews




---

This document was typeset by the author using  $\text{\LaTeX}$  2 $\epsilon$ .

The research described in this book was carried out at the Centre de Visió per Computador, Universitat Autònoma de Barcelona. Copyright © 2021 by **Hassan Ahmed Sial**. All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission in writing from the author.

ISBN: Pending

Printed by Ediciones Gráficas Rey, S.L.

"Imagination is more important than knowledge."  
— Albert Einstein

Dedicated to my parents.





# Acknowledgements

PhD is a long journey with many challenges, which is not possible to finish without the support of other people. I have always been blessed to be surrounded with supportive people, who were always there to support me in every aspect of my life. Specifically, they helped me a lot during the entire duration of my PhD. I would like to express my sincere gratitude to all of them for their trust and encouragement.

First of all, I would like to gratefully and sincerely thank my supervisors Dr. Maria Vanrell and Dr. Ramon Baldrich, for giving me the opportunity to do a PhD and also for their guidance, understanding and patience during this journey. Without their support this thesis would have never seen the light. Thanks a lot for all your encouragements during this path.

Moreover, I would like to thank Dr. Robert Benavente for all his support and advice during early years of my PhD. I appreciate his willingness to share his knowledge, ideas with me, that have helped to improve and understand this project.

During my PhD I had the great opportunity to stay at the Computer Vision Lab (CVLab), Stony Brook University, New York. I wish to express my sincere thanks to Dr. Dimitris Samaras for accepting me in his research group. I really appreciate his personal treatment, the confidence he has put in me, and the multiple good ideas he has provided during all these years.

Many thanks to every single and each one of my colleagues and friends at computer vision center for sharing with me their precious time, scientific opinions, offering me help whenever needed and for the great memories.

I would like to thank my parents and family for their support during the entire PhD period. I would like to thank my beloved wife Irum and son Azlan, who always

## **Acknowledgements**

---

allowed me to spend extra time on PhD.

Finally, let me close with an Arabic word that expresses in a perfect way all the possible thanks to the Almighty Creator, Alhamdulillah.

# Abstract

In this thesis, we explore how to estimate the effects of the light interacting with the scene objects from a single image. To achieve this goal, we focus on recovering intrinsic components like reflectance, shading, or light properties such as color and position using deep architectures. The success of these approaches relies on training on large and diversified image datasets. Therefore, we present several contributions on this such as: (a) a data-augmentation technique; (b) a *ground-truth* for an existing multi-illuminant dataset; (c) a family of synthetic datasets, *SID for Surreal Intrinsic Datasets*, with diversified backgrounds and coherent light conditions; and (d) a practical pipeline to create hybrid ground-truths to overcome the complexity of acquiring realistic light conditions in a massive way. In parallel with the creation of datasets, we trained different flexible encoder-decoder deep architectures incorporating physical constraints from the image formation models.

In the last part of the thesis, we apply all the previous experience to two different problems. Firstly, we create a large hybrid *Doc3DShade* dataset with real shading and synthetic reflectance under complex illumination conditions, that is used to train a two-stage architecture that improves the character recognition task in complex lighting conditions of unwrapped documents. Secondly, we tackle the problem of single image scene relighting by extending both, the *SID* dataset to present stronger shading and shadows effects, and the deep architectures to use intrinsic components to estimate new relit images.

Key words: intrinsic images, reflectance, shading, illumination, CNN, ground-truth generation, character recognition, relighting



# Resum

En aquesta tesi explorem com estimar els efectes de la llum que interactua amb els objectes d'una escena a partir d'una sola imatge. Per assolir aquest objectiu, ens centrem en la recuperació de components intrínseques com ara la reflectància, les ombres o altres propietats de la llum, com el color i la posició, tot això fent servir arquitectures de xarxes neuronals profundes. L'èxit d'aquest enfocament es basa en bona part en la formació de bases de dades d'imatges grans i diversificades. Les contribucions que presentem són les següents: (a) una tècnica d'augment de dades per a l'entrenament; (b) un *Ground-truth* per a un conjunt de dades multi-il·luminant ja existent; (c) una família de bases de dades sintètiques, *SID (Surreal Intrinsic Dataset)*, amb fons molt diversos i condicions de llum coherents; i (d) una metodologia pràctica per a crear *Ground-Truths* híbrids per superar la complexitat d'adquirir escenes físiques reals de manera massiva. Paral·lelament a la creació de bases de dades d'imatges, hem construït diferents arquitectures profundes de tipus codificador-descodificador molt flexibles i que incorporen restriccions físiques dels models de formació d'imatges.

A la darrera part de la tesi, apliquem tota l'experiència anterior a dos problemes diferents. En primer lloc, creem una gran base de dades d'imatges, *Doc3DShade*, híbrid amb ombres reals i reflectància sintètica sota condicions d'il·luminació complexes, i que s'utilitza per entrenar una arquitectura de dues fases que millora la tasca de reconeixement de caràcters en condicions d'il·luminació complexa de documents arrugats. En segon lloc, abordem el problema de la re-il·luminació d'escenes a partir d'una sola imatge, això es fa ampliant el conjunt de dades *SID* per representar múltiples efectes d'ombres i estudiant diverses arquitectures profundes que inclouen l'ús de components intrínseques per millorar la generació de les re-il·luminacions.

## Acknowledgements

---

Paraules clau: imatges intrínseques, reflectància, ombrejat, il·luminació, CNN, generació de *Ground-Truths*, reconeixement de caràcters, re-il·luminació.

# Resumen

En esta tesis se explora cómo estimar los efectos de la luz que interactúa con los objetos de la escena a partir de una sola imagen. Para lograr este objetivo, nos enfocamos en recuperar componentes intrínsecos como reflectancia, sombreado o propiedades de luz como el color y la posición utilizando arquitecturas de redes neuronales profundas. El éxito de estos enfoques se basa en el entrenamiento sobre grandes bases de datos de imágenes muy diversificadas. Se presentan las siguientes contribuciones: (a) una técnica de aumento de datos para entrenamiento; (b) un *Ground-truth* para una base de datos de imágenes existente con múltiples iluminantes; (c) una familia de bases de datos de imágenes sintéticas, que llamamos *SID (Surreal Intrinsic Datasets)*, con escenas muy diversificadas y condiciones de luz coherentes; y (d) una metodología para la creación de *Ground-truth* híbridos que permiten superar la complejidad de adquirir escenas físicas de manera masiva. Paralelamente a la creación de conjuntos de datos, entrenamos diferentes arquitecturas profundas de tipo codificador-decodificador muy flexibles y que incorporan restricciones físicas de los modelos de formación de imágenes.

En la última parte de la tesis, aplicamos toda la experiencia previa a dos aplicaciones diferentes. Primero, creamos una base de datos de imágenes híbrida, *Doc3DShade* con sombreado real y reflectancia sintética bajo condiciones de iluminación complejas, que ha sido utilizada para entrenar una arquitectura en dos pasos que mejora la tarea de reconocimiento de caracteres en condiciones de iluminación complejas de documentos arrugados. En segundo lugar, abordamos el problema de la re-iluminación de escenas a partir de una sola imagen, ampliamos el conjunto de datos *SID* para poder representar múltiples efectos de sombras, y estudiamos diversas arquitecturas profundas que incluyen el uso de componentes intrínsecos para poder mejorar la re-iluminación generada.



## **Acknowledgements**

---

Palabras claves: imágenes intrínsecas, reflectancia, sombreado, iluminación, CNN, generación de GT, reconocimiento de caracteres, re-iluminación.

# Contents

<b>Acknowledgements</b>	<b>i</b>
<b>Abstract (English/Spanish)</b>	<b>iii</b>
<b>List of figures</b>	<b>xiii</b>
<b>List of tables</b>	<b>xix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Image Formation Process . . . . .	2
1.2 Intrinsic Images in Computer Vision . . . . .	7
1.3 Application of Intrinsic Image Estimation . . . . .	12
1.4 Objectives and Scope . . . . .	13
1.5 Contributions . . . . .	14
1.6 Outline . . . . .	15
<b>2 Review on Intrinsic image decomposition</b>	<b>17</b>
2.1 Intrinsic Image Datasets . . . . .	17
2.2 Intrinsic Image Estimation Methods . . . . .	24
	<b>ix</b>

## Contents

---

2.2.1	Traditional Methods . . . . .	24
2.2.2	Deep Learning Based Methods . . . . .	25
2.3	Intrinsic Image Evaluation . . . . .	29
2.4	Conclusion . . . . .	30
<b>3</b>	<b>Reflectance and Shading Estimation</b>	<b>33</b>
3.1	Introduction . . . . .	33
3.2	Ground-truth Generation . . . . .	35
3.2.1	Data Augmentation . . . . .	35
3.2.2	Real Datasets . . . . .	36
3.2.3	Combined Synthetic-Real Dataset . . . . .	39
3.2.4	Synthetic Dataset : SID1 . . . . .	50
3.3	Deep Neural Network . . . . .	53
3.4	Experiments and Results . . . . .	55
3.4.1	Experiment 1. Our dataset . . . . .	56
3.4.2	Experiment 2. MIT dataset . . . . .	57
3.4.3	Experiment 3. Sintel dataset . . . . .	58
3.4.4	Experiment 4. IIW dataset . . . . .	59
3.4.5	Experiment 5. Evaluating IUI architecture . . . . .	61
3.5	Conclusion . . . . .	62
<b>4</b>	<b>Light Source Estimation</b>	<b>69</b>
4.1	Introduction . . . . .	69
4.2	SID2 Dataset . . . . .	71

4.3	Network Architecture . . . . .	73
4.4	Experiment and results . . . . .	74
4.4.1	Synthetic Dataset . . . . .	74
4.4.2	Multi-illumination Dataset . . . . .	75
4.4.3	Natural Images . . . . .	77
4.5	Conclusions . . . . .	78
<b>5</b>	<b>Applications of Intrinsic Decomposition</b>	<b>81</b>
5.1	Removing light-effects in Documents . . . . .	81
5.1.1	Introduction . . . . .	82
5.1.2	Training Dataset: Doc3DShade . . . . .	84
5.1.3	Document Reflectance Estimation Network . . . . .	87
5.1.4	Evaluation and Results . . . . .	90
5.2	Single Image Relighting . . . . .	95
5.2.1	Introduction . . . . .	95
5.2.2	SID3 Dataset . . . . .	97
5.2.3	Network Architectures . . . . .	98
5.2.4	Results and Discussion . . . . .	103
5.3	Conclusion . . . . .	107
<b>6</b>	<b>Conclusions and Future Work</b>	<b>111</b>
6.1	Conclusion . . . . .	112
6.2	Future work . . . . .	114
6.3	List of Publications . . . . .	115

## Contents

---

6.3.1 Journals . . . . . 115

6.3.2 International Conferences . . . . . 115

**Bibliography** **129**

# List of Figures

1.1	A Realist painting by (a) Gustave Courbet and (b) Luis Egidio Meléndez.	3
1.2	DRM model with the surface and the body reflection including multiple parameters used in the model.	6
1.3	Image acquisition scheme: The final image depends on illumination source, capturing device and material properties of objects in the scene	7
1.4	Shading and Shadows depends on position of light source and camera position	8
1.5	Effects of changing light source properties: Light position and colors are specified in bottom rows for each image	9
1.6	Effects of changing reflectance properties of objects	9
1.7	Top rows shows intrinsic images as visualized by Barrow and Tenenbaum in [6] and bottom rows is a color representation by Serra in [103]	10
2.1	MIT datasets: (a-c) Images under different light condition, (d) Reflectance, (e) Shading	18
2.2	MPI Sintel datasets	19
2.3	Indoor images from IIW dataset	19
2.4	Five scenes of MIII dataset	20

## List of Figures

---

2.5	Images from ShapenNet-Intrinsic dataset . . . . .	20
2.6	Example ground-truth images from Natural Environment Dataset (NED): left(Original Image), middle (Reflectance), right (Shading) . .	21
2.7	Example images from CGIntrinsic dataset: left(Original Image), right(Reflectance)	22
3.1	Example of chromaticity rotation for color-based augmentation. . . .	37
3.2	Objects in MIT Intrinsic dataset . . . . .	38
3.3	Five scenes of MIII dataset . . . . .	38
3.4	Here we demonstrate our lab setup (a) Platform image in which a red cube is placed at the center of the platform, image is illuminated by 9 light bulbs from the top and platform is tilted towards one side (b) graphical representation of our lab setup, platform has 2 degrees freedom and it can move in both pan and tilt directions, light configuration and light orientation are shown on top, all our lights are pointing towards center of the scene. . . . .	40
3.5	(a) The image illustrates that the angle of incident light is equal to the angle of outgoing light at the specular highlight on a spherical ball (b) our Light calibration setup in which we placed 150mm ball in the center of the platform . . . . .	43
3.6	Coordinate Systems in MATLAB [81], Blender [23] and our Dataset, respectively from left to right. . . . .	43
3.7	Camera Calibration Evaluation: (a) Acquired image from real-world (b) Synthetic image with parameters from camera calibration (c) Interpolated error map based on the difference between checkerboard points in real and synthetic image . . . . .	44
3.8	Camera Calibration Evaluation: (a) Acquired image from real-world (b) Synthetic image with parameters from light calibration (c) Error map as the absolute intensity difference between the real and the synthetic image central (marked with blue rectangles in (a) and (b) .	45

3.9 Refinement of camera position: (a) Acquired image from real-world  
 (b) Synthetic image with parameters from camera calibration (step 2)  
 before simulated annealing (SA) (c) Synthetic image with parameters  
 from camera calibration (step 2) refined with simulated annealing  
 (SA) (d) A blended image of (a) and (b) with a color pattern at the  
 bottom left. (e) A Blended image of (a) and (c) with a color pattern at  
 the bottom left. . . . . 45

3.10 Refinement of light position: (a) Acquired image from real-world  
 (b) Synthetic image with parameters from light calibration (step 2)  
 after simulated annealing (SA) (c) Error map as the absolute intensity  
 difference between the real and the synthetic image central (marked  
 with blue rectangles in first two images) (d) Acquired image of a sphere  
 (e) Synthetic image of a sphere . . . . . 46

3.11 Example of GT data with our lab setup using MIT intrinsic[44] dataset  
 creation technique: (a) Image captured in real (b) white Shading is  
 captured in real (c) GT Reflectance is created by dividing color image  
 with white shading . . . . . 47

3.12 Example GT images from DoC3DShade Dataset: (a) Shading with doc-  
 ument Material is captured in real (b) reflectance texture is rendered  
 in the synthetic world (c) and GT image is generated by multiplying  
 acquired with synthetic. . . . . 48

3.13 Example GT images from our combined synthetic-real Dataset: (a)  
 GT Image is captured in real (b) shading image is rendered in the  
 synthetic world (c) GT Reflectance is generated by dividing acquired  
 image with rendered shading. . . . . 49

3.14 Multi-view Stereo: (left) set of images from different viewpoint, (mid-  
 dle) sparse reconstruction by using structure from motion algorithms  
 (right) Dense reconstruction from initial sparse reconstruction. We  
 used Visual SFM [36, 127] to get 3D reconstitution in this image. . . . 49

3.15 Photometric stereo: (a) input images with calibrated lights and camera  
 (b-d) reflectance, shading and shape estimation with [98] . . . . . 50

3.16 Dataset Generation Setup . . . . . 52



## List of Figures

---

3.17 IUI-Network architecture. One encoder and two decoders for reflectance and shading estimation. Three inter-related loss functions. Type of layers are indicated by a color code given at right-bottom of the figure. Scheme of inception modules are given at left-top of the figure. . . . .	55
3.18 Some examples of our SID dataset. (a) Original Images. (b) and (d) are Reflectance and Shading estimation by our IUI network, respectively. (c) and (e) are GT Reflectance and Shading, respectively. . . . .	64
3.19 Qualitative Results on MIT intrinsic image dataset, compared to other methods, we achieved sharp and better colors and removed shading effects. Our method performed best in bringing reflectance details from dark part of image. . . . .	65
3.20 Visual comparison on MPI-Sintel dataset using image split. . . . .	66
3.21 Visual comparison on MPI-Sintel using Scene split. . . . .	67
3.22 Qualitative results on IIW . . . . .	67
4.1 Image Generation Setup. Camera and light positions are given in spherical coordinates $(r, \theta, \varphi)$ . . . . .	72
4.2 Deep Architecture. Inception module from [116] . . . . .	74
4.3 Direction and Color estimation examples on SID2 dataset: (a) Original images, (b) Generated images with estimated light properties, (c) RGB Image subtraction between (a) and (b). Bottom rows are the corresponding computed errors for direction and color in degrees, ordered from smaller (left) to larger (right) direction estimation error. . . . .	78
4.4 Direction and Color estimation examples on Multi-illumination dataset: (a) Original images, (b) Ground-truth plotted on corresponding spheres, (c) Estimations provided by our proposed architecture. Bottom rows are computed errors for direction and color in degrees. . . . .	78
4.5 Examples of light direction estimation on natural images. Predicted direction is plotted top left in each image. . . . .	79

5.1 OCR accuracy (by Tesseract[88]) comparison on document images pre-processed with our proposed method vs. Kligler *et al.* (state-of-the-art document shadow removal method[62]) vs. the commercially available document capturing application CamScanner. . . . . 83

5.2 Data Creation Pipeline: (a) Shows the hardware setup. The captured shapes are textured in Blender and combined with the captured shading image by element-wise multiplication to create  $I$ . The  $\otimes$  denote the element-wise multiplication. The ‘orange’ and ‘blue’ arrows denote the rendering and the combining. . . . . 85

5.3 Doc3DShade capture setup modified from our original setup (figure 3.4), We removed Central light and introduced Intel RealSense RGB Depth CameraD435 camera on top to acquire shape and shading images. . . . . 86

5.4 Proposed framework: The WBNet takes RGB image,  $I$  as input and produces the white-balance kernel ( $\hat{W}B$ ). The white-balanced image,  $\hat{I}_{wb}$  is then forwarded to the SMTNet which regresses the material ( $\hat{M}$ ) and the shading,  $\hat{S}_p$ . The  $\otimes$  denote the element-wise multiplication, ( $/$ ) denote division and the triangles denote the loss functions. . . . . 88

5.5 Qualitative results on real images after applying white balancing (After WB) and shading removal (After Shd. Rem.). Input images are non-uniformly illuminated with two lights. . . . . 92

5.6 Results on real-world images from [78]. [26]’s shading removal method fails to retain the background since shading, illumination and document background is modeled as a single modality. . . . . 93

5.7 Comparison with existing shadow removal methods [3, 56, 62, 121] on real image sets: (a), (b), (c) are obtained from [3, 56, 121] respectively. These comparisons show our method well generalizes on soft shadows. We report a fail case on hard shadows at the bottom row of (c). . . . . 94

5.8 (a) Comparison with IIW method [12], IIW fails to accurately preserve text. (b) Results on multi-illuminant OCR dataset [84], WB is white-balanced image and SMR is output after removing material and shading. 94

## List of Figures

---

5.9 Single Image Relighting: generation of relit versions of the original image in the center for 8 different light on-top positions in front-right, front-top, front-left, left, right, back-right, back, back-left (from left to right and top to bottom). . . . .	96
5.10 Synthetic world for image generation . . . . .	98
5.11 Some examples of the SID3 dataset. For 2 scenes, the reflectance component is on the right column and 5 different light conditions are shown from left to right. For each light condition we show the image (top row) and its corresponding intrinsic shading (bottom row). . . .	99
5.12 Proposed network architectures: (a) 1-to-1 U-Net with one encoder and one decoder (b) 1-to-2 intrinsic with one encoder and two decoders (c) 1-to-3 intrinsic with one encoder and three decoder . . . .	100
5.13 A full example of the results on SID3 dataset with all the intrinsic components. . . . .	105
5.14 Two complex examples on SID3 dataset. . . . .	105
5.15 Qualitative results on VIDIT dataset, images from (a)-(f) are obtained from [47], (g) our prediction with 1-to-1 U-Net . . . . .	108
5.16 Qualitative results on real images . . . . .	108

# List of Tables

2.1	Comparison on current available dataset according to several properties. From left to write we account for: Number of images, ground-truth (GT) perfectly fulfills the physical model, GT is on the full image or only a part, GT is presenting the influence of a diverse background, GT is presenting cast shadows apart from shading, and global image present physically consistent lighting. Meaning of special cases: (★) MIT and MIII datasets generally fulfills product model by including a factor i.e. $I = \alpha(R \cdot S)$ , but it does not completely hold for all images and have small deviation; (‡) Sintel dataset present diverse backgrounds compared to the rest, but with a strong bias towards specific colors due to high correlation of a video sequences. (†) Training area is large, but still does not cover the full image. . . . .	23
2.2	Comparison of tradition intrinsic image decomposition methods based on inputs, physical cues used and outputs. Other physical cues (1) higher level color descriptor (2) shape cues (3) texture structure	26
2.3	Comparison of deep learning intrinsic decomposition methods based on inputs, type of deep neural network and outputs. Other outputs (1) Shape (2) Normal Map (3) Illumination model . . . . .	29
3.1	Errors for reflectance and shading predictions on our dataset. Comparison between our IUI architecture and Retinex algorithm. IUI decreases the error of Retinex by the factor given in brackets. Errors are separately reported on object, on foreground and the on whole image. . . . .	57

## List of Tables

---

3.2	Estimation errors on MIT dataset reported in previous works by different methods and for our IUI architecture. . . . .	58
3.3	Results on Sintel Image Split dataset. Best errors are highlighted in bold.	60
3.4	Result on Sintel Scene Split dataset. Best errors are highlighted in bold.	60
3.5	Result on IIW dataset . . . . .	61
3.6	Estimation errors of different architectures trained on Shapenet-Intrinsic dataset. In bottom row the errors of IUI architecture trained on our dataset and tested on Shapenet-Intrinsic. . . . .	62
4.1	Estimation Errors (in degrees) for light source direction and color with the proposed architecture trained on SID1 and SID2. . . . .	75
4.2	Estimation Errors (in degrees) for light direction and color at different tilt levels. . . . .	75
4.3	Estimation Errors (in degrees) for light direction and color at different pan levels. . . . .	76
4.4	Estimation errors in degrees on two versions of MID dataset (with Masked or UnMasked spheres). . . . .	76
4.5	Estimation errors in degrees dividing MID dataset in front and back light. . . . .	77
5.1	Quantitative comparison of [26]’s unwarping quality on DocUNet benchmark dataset [78] when our proposed approach is applied as a pre-processing step before OCR. . . . .	91
5.2	Quantitative results of SID3 Dataset . . . . .	106
5.3	Quantitative Results of VIDIT dataset . . . . .	107

# 1 Introduction

Computer vision is an interdisciplinary scientific field that computationally models all steps involved in human visual system to empower computers to capture, process, analyze and understand images. The digital image is the starting point to depict real-world scenes in the computer vision process. Each image value represents the physical interaction between the illumination source and the scene objects which provokes some complex effects such as shading, shadows, specularities and inter-reflectances. These effects mainly depends on the position and color of the light, the position and materials of the objects and on the whole scene geometry. Extracting and isolating these effects from the images is known as the intrinsic image decomposition problem [6], and it is going to be the focus of this thesis. Intrinsic image decomposition is considered to be a challenging problem in the research community and it has applications in both computer vision and computer graphics fields; examples varies from segmentation or shadow removal, to re-lighting or style transfer.

Recent improvements in computational power and innovations in artificial intelligence resulted in the success of deep learning based method to resolve different computer vision tasks. Deep Convolution Neural Networks (CNN) have been extensively used in the past decade across different domains in imaging related tasks, and in particular, many researchers tried to resolve intrinsic image decomposition problem with deep learning based approaches. However, in doing so, the main difficulty has been the lack of large and realistic image datasets to train the CNN architectures, which has been the bottleneck in this field.

Therefore, In this thesis, we explore deep architectures incorporating physi-

cal constraints for intrinsic image estimation, and we provide a family of surreal datasets with strong light effect interactions to solve the complexity of building real datasets. We also put some basis to build combined synthetic-real image ground-truths (GT)<sup>1</sup> to solve specific problems. Overall, we show the effectiveness of theoretical concepts of intrinsic image to solve real-world applications.

In this chapter, we give a brief overview of how intrinsic image estimation has been modelled in computer vision, as well as the challenges and applications associated with this problem, which are the basis for the rest of the work. At the end, we list the major contributions of this thesis and give an outline of the thesis organization.

### 1.1 Image Formation Process

An image is a medium to provide a visual representation of a real-world scene and it is generally a two-dimensional photograph, painting, or drawing that resembles physical objects and natural scenes. Images have always remained part of our life to store and communicate visual information.

In the field of visual arts, artists use visual clues such as occlusion, relative position or scale and advanced techniques such as a change in illumination (*i.e.* shading and cast shadows), variation in texture and transparency to give a sensation of depth and volume to give global realism to the painting. Figure 1.1, shows famous paintings by (a) Gustave Courbet and (b) Luis Egidio Meléndez, in which most of the above mentioned techniques essentially shading and cast shadows are used to bring naturalness to the projected scene. In these paintings, the physical reflection of the light on object's geometry and textures is projected in the form of shading and cast shadows, all combined in one single image. It is interesting to note that humans develop the ability to use these visual cues to isolate specific physical properties of the scene by just watching the image. For instance, we can estimate the natural light position in the scene, understand shape, textures and material properties of the objects. These physical properties are an intrinsic part of the images. As mentioned earlier, the task of computer vision is to mimic the human visual system in computers, and the task of isolating these physical properties from a two-dimensional image is known as the intrinsic image decomposition problem. In the next paragraphs, we will further explain the concepts of intrinsic images for digital image formation.

---

<sup>1</sup>The terms ground-truth and GT are used indistinctly in this thesis



(a)



(b)

Figure 1.1: A Realist painting by (a) Gustave Courbet and (b) Luis Egidio Meléndez.

The formation of images involves some radiometric and geometric processes to map the three-dimensional world to a two-dimensional space. The digital image is a two-dimensional array of a finite set of digital values known as image pixels. Each of these pixels captures the result of a physical interaction of light with the scene's objects. There are many factors that influence the final image pixel values, but the most important of these are : (a) position and type of light source used to illuminate the scene; (b) object shapes, surface reflecting properties and scene geometry; and (c) the image capturing device. All these parameters have been formalized in different image formation models in the fields of computer graphics and computer vision [91].

Shafer presented a light reflection model called Dichromatic Reflection Model (DRM) [106], which has been one of the most used in computer vision. This model defines the radiance of the light reflected from the object,  $L_{obj}$ , as the addition of two light components, surface reflection,  $L_s$ , and body reflection,  $L_b$ .

$$L_{obj}(\theta_i, \theta_r, \theta_g, \lambda) = L_b(\theta_i, \theta_r, \theta_g, \lambda) + L_s(\theta_i, \theta_r, \theta_g, \lambda) \quad (1.1)$$

where  $\theta_i$  is the angle between the incident light vector,  $\mathbf{V}_i$ , and the object surface normal,  $\mathbf{N}$ .  $\theta_r$  is the angle between the reflected light vector  $\mathbf{V}_r$  and the normal  $\mathbf{N}$ . Finally,  $\theta_g$  is the phase angle between vectors  $\mathbf{V}_i$  and  $\mathbf{V}_r$  (figure 1.2 illustrates the above-mentioned photometric angles). The first three angles can be grouped in a single set of parameters that represents the scene geometry denoted as  $\Theta$ , whereas



$\lambda$  keeps isolated to represent the spectral<sup>2</sup> properties of the reflected light. The assumptions of the model which are opaque, non-fluorescent dielectric materials with single light source and no inter-reflections, amongst others [106] makes also to conclude that each of these light components can be described in a quite realistic way as the product of two independent terms, a geometric factor,  $m_s(\Theta)$  or  $m_b(\Theta)$ , which only depends on the geometry of the scene; and a relative spectral power distribution  $c_s(\lambda)$  or  $c_b(\lambda)$ , which only depends on the spectral component of the reflected light and not on the geometry. Replacing these terms in the equation 1.1 yields to:

$$L_{obj}(\Theta, \lambda) = m_b(\Theta)c_b(\lambda) + m_s(\Theta)c_s(\lambda) \quad (1.2)$$

Equation 1.2 gives continuous domain representation of the reflected light from the object  $L_{obj}$ . When this reflected light enters into the camera capturing device, it goes through the *spectral integration* process which involves filtration, sampling and integration steps on visible light spectrum<sup>3</sup> to obtain the digital image  $I$ . [64].

$$I(x, y)_k = \int_{\omega} L_{obj}(\Theta, \lambda) \mathcal{S}_k(\lambda) d\lambda \quad (1.3)$$

where  $(x, y)$  denotes the spatial representation in the image plane that is determined by  $\Theta$  and objects position with respect to camera,  $\mathcal{S}_k$  is the sensor responsivity through the visible light spectrum,  $\omega$ . Each capturing device uses certain numbers of sensors usually three with variable spectral response that leads to different image representations of the same scene. Substituting 1.2 equation into equation 1.3, we obtain:

$$I(x, y)_k = m_b(\Theta) \int_{\omega} c_b(\lambda) \mathcal{S}_k(\lambda) d\lambda + m_s(\Theta) \int_{\omega} c_s(\lambda) \mathcal{S}_k(\lambda) d\lambda \quad (1.4)$$

The standard DRM model assumes that that surface reflectance has neutral interface reflection or constant interface reflection  $h$  which is independent of  $\lambda$  and its surface power distribution  $c_s(\lambda)$  is mainly dependent to incident light power

---

<sup>2</sup>The spectral power distribution, spectrum, of a light source is the concentration of power per wavelength unit, which is responsible of the light perceived color.

<sup>3</sup>Visible light spectrum covers the range of electromagnetic spectrum which is visible to human eye, primarily its the range human can see colors and its is approximately between  $400nm$  (violet) to  $700nm$  (red)

distribution  $e(\lambda)$  [63, 119] which appears as highlight on object surface:

$$c_s(\lambda) = he(\lambda) \quad (1.5)$$

But the body spectral power distribution  $c_b(\lambda)$  depends on both, the incident light power distribution  $e(\lambda)$  and the body reflectance or albedo function  $g_b(\lambda)$  and it exhibits the properties of object color and object shadings.

$$c_b(\lambda) = g_b(\lambda)e(\lambda) \quad (1.6)$$

So, we can rewrite equation 1.4 as:

$$I(x, y)_k = m_b(\Theta) \int_{\omega} g_b(\lambda)e(\lambda)\mathcal{S}_k(\lambda)d\lambda + m_s(\Theta) \int_{\omega} he(\lambda)\mathcal{S}_k(\lambda)d\lambda \quad (1.7)$$

Usually, the infinite vector space of the spectral integration is reduced to a three dimensional vector space, the red, green and blue color space, by analogy with the human color representation based in three types of cones. Each color pixel  $(r, g, b)$  in the image  $I(x, y)$  represents the camera color response to the reflected light from the object. The transformation from infinite vector space to three dimensional space is linear [105], that defines the DRM model as the linear combination of two linear color vectors:

$$I(x, y) = m_b(\Theta)\mathbf{C}_b + m_s(\Theta)\mathbf{C}_s \quad (1.8)$$

$\mathbf{C}_b = (r_b, g_b, b_b)$  and  $\mathbf{C}_s = (r_s, g_s, b_s)$  are the colors of the body and surface reflection on the object. In this equation 1.8, the first term  $m_b(\Theta)\mathbf{C}_b$  is the body reflectance that accounts for the light reflected after interacting with surface albedo and the second term  $m_s(\Theta)\mathbf{C}_s$  describes the light reflected directly from the object surface. Figure 1.2 demonstrates the effects of the body and the surface reflectance: body reflectance creates the diffuse parts in the image and surface reflectance is responsible for specularities in the image.

Figure 1.3 shows the image formation process in which light emitted from illumination source propagates through the scene and interacts with the objects and ultimately reflects into the camera sensor. As this light interacts with scene objects it creates different light effects. Cast shadows are produced when an opaque object is placed in the direction of a light source. While shading represents a gradual change in the brightness level of a surface coinciding with a gradual change of its

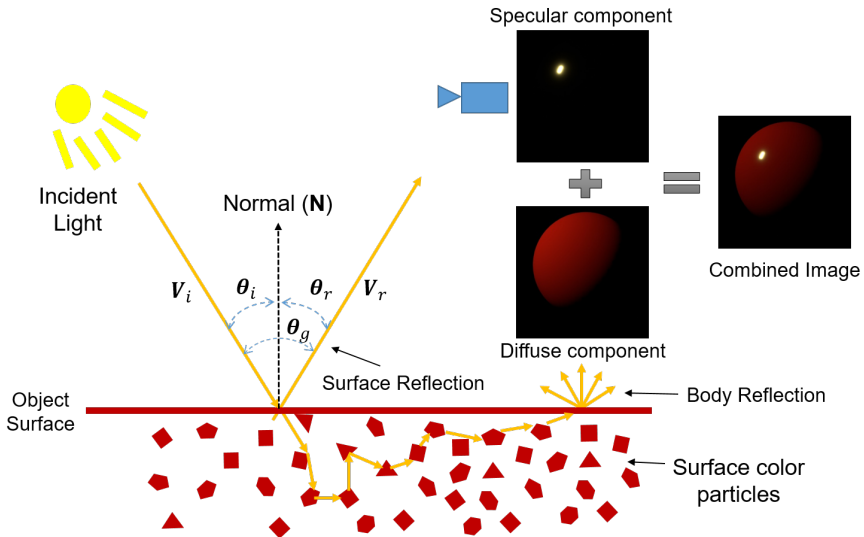


Figure 1.2: DRM model with the surface and the body reflection including multiple parameters used in the model.

normal directions.

Figure 1.4 demonstrate how shading and shadows are formed at different position of the image by changing light and camera position. The color of the light source affects the final color of an image pixel. The light reflected by the object surfaces themselves are acting as new light sources projected on the rest of the scenes, provoking what is know as inter-reflections.

The light reflected by the object surfaces also depends on the material properties. There are broadly two types of materials present in nature: dielectric and conductive.

Examples of dielectric materials are wood, paper, ceramic, and plastic and examples of conductive materials include any type of metal i.e. copper, gold, and steel. As shown in figure 1.2 when an incident light ray touches the object surface, part of its energy is absorbed and the other is reflected from its surface. The light reflect in either of two ways: specular reflection is mirror like reflection from object surface according to surface normals, diffuse reflection is scattering of absorbed light in all the directions. The Dielectric materials have both specular and diffuse

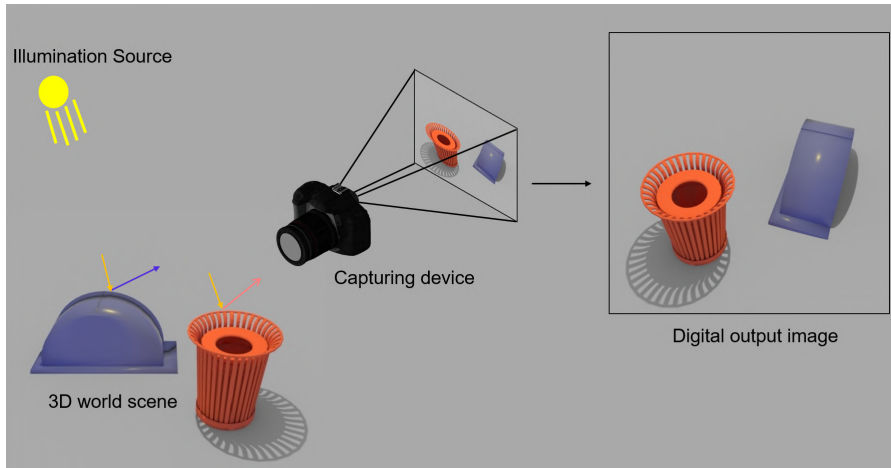


Figure 1.3: Image acquisition scheme: The final image depends on illumination source, capturing device and material properties of objects in the scene

reflectance properties while the metallic surfaces are very reflective with no diffuse component. Object true color or its texture is known as albedo or *reflectance* in research, this only depends on the material and independent of scene illumination and camera viewpoint.

Figure 1.5 shows different images by changing illumination source color and direction. This illumination variation in each image results in different shading and cast shadows position on final digital images. Likewise, figure 1.6 illustrate the effect of changing the object's color or texture. In these images, we can also visualize that the lower part of the right basket is also illuminated by inter-reflected light from the left object.

## 1.2 Intrinsic Images in Computer Vision

The "*intrinsic image*" term in computer vision was first introduced by Barrow and Tenenbaum [6]. In this work, the authors suggested that an intensity image can be decomposed into a family of intrinsic characteristic images. The intrinsic properties they mainly focused on were incident illumination or shading, reflectance, and distance or surface orientation. They also mentioned other intrinsic characteristics such as specularity, luminosity and transparency. The input for this problem is one

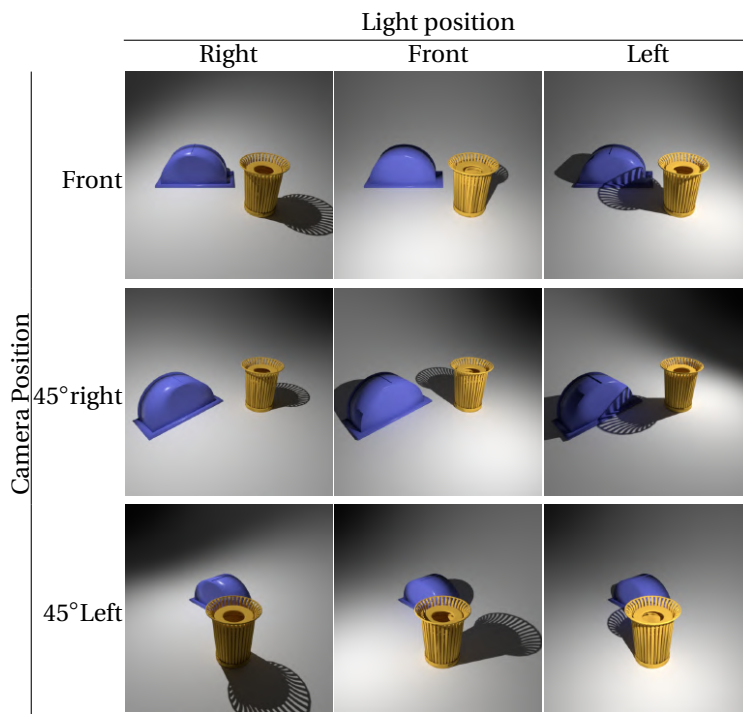


Figure 1.4: Shading and Shadows depends on position of light source and camera position

or multiple images of the same scene and the objective was to recover a family of output images, each representing individual characteristics. The central problem for such a decomposition is that an intensity image is already confounded since each intensity value encodes information of all of these intrinsic characteristics of the image at the corresponding scene point. The support of their idea came mainly from the fact that humans can disentangle these physical characteristics from any visual scene. Figure 1.7 top-row shows how authors visualized these intrinsic images almost four decades ago [6], bottom-row is a color representation of these original images taken with permission from Serra's Ph.D. dissertation [103].

Barrow and Tenenbaum [6] proposed a computational model to estimate these intrinsic images based on edges in the intensity image. The authors did several

## 1.2. Intrinsic Images in Computer Vision

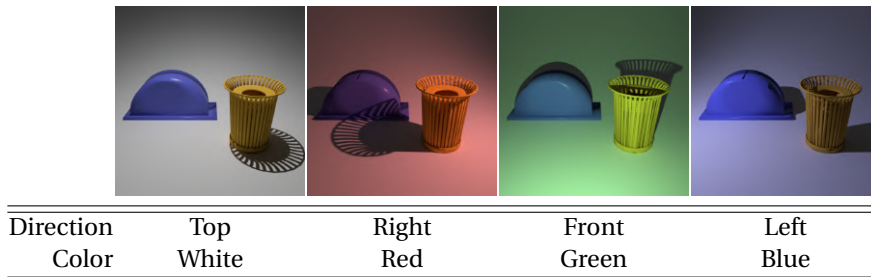


Figure 1.5: Effects of changing light source properties: Light position and colors are specified in bottom rows for each image

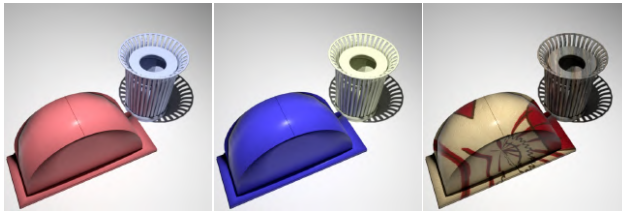


Figure 1.6: Effects of changing reflectance properties of objects

assumptions about the scene objects, illumination source, capturing system and encoding process. They defined a simplified world satisfying the following requirements:

- Objects surfaces are relatively smooth with gradual distance and orientation variation over its surface. There are no sharp edges.
- Objects surfaces are lambertian, it means that light reflects in all direction and there are no specularities in the scene.
- Objects are only illuminated by distant illumination source with known direction and power. Inter-reflectance (light reflected from nearby objects) is also assumed to be minimal.
- Capturing sensor noise and quantization effects can be ignored.

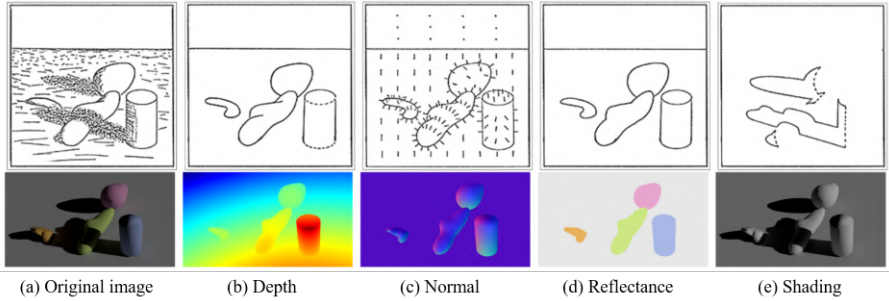


Figure 1.7: Top rows shows intrinsic images as visualized by Barrow and Tenenbaum in [6] and bottom rows is a color representation by Serra in [103]

- Scene is captured from a general position and only one single viewpoint of the scene is available.

Images captured under these assumptions have continuously varying intensity regions with step discontinuity at the object's edges. The authors studied the relationship between these regions and boundaries to estimate intrinsic properties.

Subsequent authors also did similar assumptions about scene, objects and capturing system (we review these methods in chapter 2). The most widely used hypothesis in intrinsic image decomposition research is lambertian surfaces and white canonical light in the scene that remove surface reflectance component ( $m_s = 0$ ) from the DRM model (equation 1.7) and based on this an image can be decomposed into reflectance and shading components:

$$I(x, y) = R(x, y) \cdot S(x, y) \quad (1.9)$$

where  $R$  and  $S$  denote the reflectance and shading components, respectively, and  $(x, y)$  represents a pixel coordinates of the image,  $I$ . *Albedo or reflectance* accounts for reflection of light from object surface and it is material-dependent only while *shading* represents brightness variation in the image due to objects geometry and inter-reflectance.

When lambertian assumption about object surface is relaxed, it incorporates the surface reflectance component from the DRM model (equation 1.7) and the

intrinsic image model can be formulated as [4, 68, 110]:

$$I(x, y) = R(x, y) \cdot S(x, y) + Sp(x, y) \quad (1.10)$$

$Sp$  accounts for the specular reflection of the scene and is modeled as a new term added to equation 1.9.

Barron and Malik [4] proposed a model in which the shading term is represented as a function  $F$  of shape ( $Sh$ ) and of the source illumination component,  $L$ , which leads to the formulation as:

$$I(x, y) = R(x, y) \cdot F(Sh(x, y), L) \quad (1.11)$$

In their model the illumination properties of the scene are intrinsic components  $L$  which represent both color and direction of the light source, they represented illumination properties in the form of a light probe.

Finally, Serra [103] proposed an intrinsic model based on the DRM (equation 1.7) that also includes the camera sensors and scene illumination terms. They proposed that a pixel value at an image can be isolated into effects of the camera sensor and illumination components given by  $3 \times 3$  matrices and are denoted as  $\mathbf{S}_{sen}$  and  $\mathbf{L}_{col}$  respectively.

$$I(x, y) = \mathbf{L}_{col} \mathbf{S}_{sen} (R(x, y) \cdot S(x, y) + Sp(x, y)) \quad (1.12)$$

In this dissertation, we mainly focused on three intrinsic image properties namely: reflectance, shading and illumination. First, we concentrated on a more basic intrinsic image model to estimate reflectance and shading from a single image according to equation 1.9. We used an end-to-end deep learning based approach with a physical loss to estimate reflectance and shading from a single image. (more details on this are provided in chapter 3). Secondly, we worked on estimating illumination properties from a single image in chapter 4. These illumination properties were the light color and the light position from a single image. We used a light representation slightly different from the one used by Barron and Malik [6]. We used the form of numerical values rather than a light probe. Our light color representation is more similar to Serra [103] illumination components  $\mathbf{L}_{col}$ , we represented light color on three  $RGB$  numbers and light direction is measured in a spherical coordinate system with reference to camera position. Finally, in chapter 5, we demonstrated practical applicability of these intrinsic image models. In the



first application we created a large *Doc3DShade* dataset to remove shading effects from the document images. This dataset combines natural paper material with synthetic texture. In the second application we used all three intrinsic properties of reflectance, shading and illumination for single image relighting task.

To conclude the section, we can sum up by saying that intrinsic image decomposition is an inverse optics problem to estimate light-dependent properties, such as *shading* and *illumination*, along with material-dependent properties, such as *reflectance* or *albedo*.

### 1.3 Application of Intrinsic Image Estimation

Intrinsic image decomposition has been used for different applications we can find in the computer vision and computer graphics literature:

- Image Re-coloring or re-texturing by modifying the reflectance component only while preserving illumination coherence to change appearance of the scene. [11, 15]
- Intrinsic colorization to change a gray-scale image to color image by applying color transfer in reflectance image which is illumination invariant [75].
- Image based relighting to change the shading component only while keeping true color of object intact in the scene [5, 15, 30].
- Estimating illumination component helps in computer graphics application to introduce new objects in the scene with the coherence with the shading and shadows to better blend in the image [40, 73].
- Face relighting or portrait relighting, that estimates the source light and illuminates a new face image with the target light [90, 115, 135].
- Shadow removal is used with intrinsic image estimation to detect shading and shadow free roads for autonomous driving and surveillance [65, 83]. While for the document images, shading removal can increase the overall accuracy of document digitization applications [27].
- Shape from shading applications are required to estimate both shading and illumination direction in the images to transform it to 3D shape [5, 48].

## **1.4 Objectives and Scope**

One of the aims of this research is to explore deep architectures and visual cues related to intrinsic image estimation. To achieve this goal we studied how intrinsic image components namely: albedo, shading, illumination and relations of these can be processed with convolutional neural networks. The objective of this research is to design deep architectures to estimate all forms of intrinsic images to boost the results of different computer vision applications.

In parallel with the previous aim, we also focus on the generation of large ground-truths with intrinsic components. Convolutional neural networks require a huge amount of data to be trained, making datasets for this kind of application require complete controlling of lights and careful selection of objects and its materials which is only possible in laboratory conditions. Currently available datasets have two main types of problems: either the data is realistic in a photometric way, but the amount of available data is low, or the data is synthetic and consequently less realistic, but the amount of data can be high. Therefore, a second objective of this work is to develop datasets for intrinsic image decomposition by resolving the aforementioned problems. We have done a major work in getting such large datasets and we also provided an extended review of current available dataset in chapter 2.

In chapter 3 we focus more about challenges associated with ground-truth generation for the intrinsic image decomposition and its achievable solutions.

First, we proposed a color-based data augmentation technique that extends the training data by increasing the variability of chromaticity and preserving the reflectance geometry of the ground-truth. In this way the lack of data can be partially solved with data augmentation.

Secondly, We proposed a pipeline to build a dataset by registering synthetic with acquired scenes to be able to automatize the process of building the dataset ground-truth. This is an ambitious objective that made us face a list of problems like: setting a platform with pan-tilt movement where to create the scenes, estimating light and camera physical positions, representing the physical world in a Computer Graphics render engine, apply algorithms of camera calibration to convert from camera coordinates to the physical word and then computing errors between synthetic and acquired images. Finally, we introduced a completely synthetic dataset of 25,000 images named SID for *Surreal Intrinsic Dataset*[112]. This dataset has a lot of variation of shading and reflectance effects, and jointly with it, we proposed a new deep learning architecture to predict both shading and reflectance in parallel.

We used a new loss function based on the basic definition of Intrinsic images. The performance of the proposed architecture trained on our synthetic dataset was very promising. In chapter 3 we showed results for intrinsic image estimation using our deep architecture on synthetic and other datasets.

In chapter 4 we focus on another intrinsic image component by estimating light source properties from a single image and how to use it for more complex tasks such as scene relighting. As an initial step in this domain, we presented a method to estimate the direction and color of a scene light source from a single image. This work is based on two main ideas: (a) we use a new synthetic dataset with strong shadow effects with similar constraints to *SID dataset*[112]; and (b) we define a deep architecture trained on the mentioned dataset to estimate direction and color of the scene light source. Apart from showing a good performance on synthetic images, we additionally propose a preliminary procedure to obtain light positions of the Multi-Illumination dataset[86], and, in this way, we also prove that our trained model achieves a good performance when it is applied to real scenes.

Finally, we showed how intrinsic decomposition can be used for two different computer vision applications in chapter 5. In the first application, we made a dataset based on our combined synthetic-real dataset creation methodology of chapter 3 to enhance results for automatic document content processing which are mostly affected by artifacts caused by the shape of the paper, and non-uniform and diverse color of lighting conditions. We collected a large-scale Document dataset, Doc3DShade, which combines diverse, realistic illumination scenarios with natural paper textures. In the second application we combined our ideas of intrinsic decomposition in reflectance and shading and light estimation from single image for image relighting task. We modify shading properties of image to get new relighted image based on new illumination settings. We extended our SID2 dataset by extending number of objects in each scene to provoke more shading and shadows cues for this task and capturing same scene under multiple illumination conditions.

## 1.5 Contributions

The contributions for this thesis are the following:

- A trained inception based architecture with double loss-function to predict both shading and reflectance in parallel.

- A family of surreal synthetic image datasets (SID1, SID2 and SID3) presenting a wide range of variations of light conditions and interactions.
- A new technique for data augmentation based on color chromaticity rotation adapted for intrinsic decomposition.
- An architecture to estimate light properties, position and color, from a single image, as well as the creation of a ground-truth for the Multi-illuminant dataset [86].
- A combined synthetic-real image dataset of photometrically realistic images of documents where the real ground-truth is acquired for shading and synthetically generated for reflectance.
- A large study of applying different architectures on the Single Image Relighting problem using a synthetic ground-truth.

## 1.6 Outline

The chapters of this thesis dissertation have been organized as follows:

**Chapter 2:** This chapter provides extensive overview of current available datasets and methods for intrinsic image estimation. We summarized existing datasets according to several properties in table 2.1. We gave a brief overview of existing methods on single image intrinsic decomposition before and after the deep learning appearance.

**Chapter 3:** This chapter focuses on getting basic intrinsic image properties of reflectance and shading from single image. In this chapter, we discuss in detail challenges associated with ground-truth generation for this task and how we resolved this challenge by first introducing data augmentation techniques and then by introducing our own dataset generation pipelines. In this chapter, we proposed a deep neural network with physical constraints to estimate reflectance and shading intrinsic decomposition.

**Chapter 4:** In this chapter, we estimate another intrinsic modality to know light source direction and color from a single image. We extended our dataset generation pipeline of the previous chapter to generate a ground-truth for this task and we introduce a deep learning based approach to estimate the scene light properties from a single image.

**Chapter 5** In this chapter, we demonstrated how theoretical concepts of intrinsic image estimation can be used for two different practical applications. First, we removed shading effects from document images for better document digitization, in this task we build a document image dataset having a variety of shape and shading effects on document images and then we developed a deep learning model to estimate shading free reflectance image. In the second application, we extended our synthetic dataset generation pipeline of chapter 4 to accommodate it for the image relighting task. We designed deep neural network that estimate reflectance and shading components of source and target light and use physical losses based on intrinsic decomposition to predict relighted image.

**Chapter 6** This chapter presents the overall summary of the ideas developed in this dissertation and gives possible future directions based on the thesis contributions.

## 2 Review on Intrinsic image decomposition

The aim of this chapter is to review the previous work on intrinsic image decomposition. We have divided this revision in two parts. Firstly, we review the datasets developed for this task and we analyze their main pros and cons. Secondly, we survey the methods that have been developed to predict the intrinsic components.

The main identified problems regarding the datasets for intrinsic image estimation swing between the realism of the physical lighting properties of the scene, and the number of images they provide to be enough to train deep architectures. We will start from the pioneering MIT intrinsic dataset [44] formed by 20 different objects. Regarding the different methods developed for intrinsic estimation we conclude that the performance of trained architectures usually relies on a large number of training samples and a high variable appearance between them.

### 2.1 Intrinsic Image Datasets

*MIT Intrinsic*[44] was the first dataset in this domain, and it is a non synthetic dataset that was captured under very controlled conditions. It has 20 objects, captured in 11 different lighting conditions, although only one shading and one reflectance pair is provided in the final dataset ground-truth. They used a gray-spray technique to create ground-truth intrinsic images in which first they captured a non-painted color object image under 11 different lighting conditions, and then the object is painted with gray spray and placed at the same location and captured again under the same illuminations. Gray painted object image is considered as

shading ground-truth and the reflectance ground-truth is obtained by dividing the non-painted image with the painted one supported by the physical model of equation 1.9. Different intrinsic image versions can be generated by introducing a scalar  $\alpha$  that minimizes the difference,  $I(x, y) - (\alpha R(x, y) \cdot S(x, y))$ , since even in such controlled environment the product model does not hold for all pixels in objects. The ground-truth is just provided for the pixels in the object mask. Two example images from MIT dataset are shown in figure 2.1.

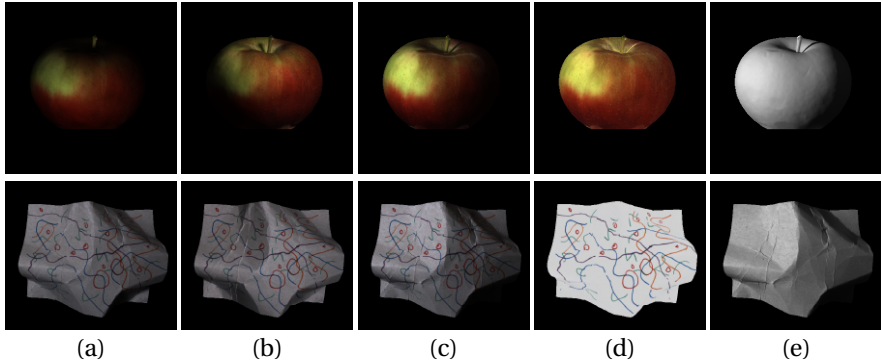


Figure 2.1: MIT datasets: (a-c) Images under different light condition, (d) Reflectance, (e) Shading

*MPI Sintel* [17] is a synthetic dataset based on an animated movie. It has 18 scenes with 50 frames each, except one that has 40, totaling 890 images. Sintel was the first large dataset giving the opportunity to train deep architectures in the last years. However, it presents unnatural shading, and some color bias mostly on blue and brown colors. Thus, the generalization of networks trained on Sintel are affected by these color biases. And, like in all synthetic datasets, some erroneous pixels are found around boundary of objects. Figure 2.2 shows ground-truth images from MPI Sintel dataset.

*IIW: Intrinsic Images in the Wild* [12] is a large and realistic dataset of 5230 images (see figure 2.3). It only provides sparse reflectance pairwise judgments as training data. These judgments does not present a spatial coherent map for reflectance. Consequently, networks trained using this dataset present too smooth reflectance estimation with lack of texture variation.

*MIII dataset* [10] is an extension of the MIT Intrinsic to a multi view multi

## 2.1. Intrinsic Image Datasets



Figure 2.2: MPI Sintel datasets



Figure 2.3: Indoor images from IIW dataset

illuminant intrinsic dataset. Dataset has 5 scenes and each scene is captured in 20 different illumination conditions with 6 cameras views giving the total of 600 images. Inspired by MIT intrinsic dataset [44], each object in the scene is gray painted and capture again to get ground-truth reflectance by dividing the color image with the gray painted image. They also provided raw depth and 3D point cloud information. Figure 2.4 shows five scenes of the MIII dataset captured by one of six cameras.





Figure 2.4: Five scenes of MIII dataset

*ShapeNet*[110] is the first large dataset in this field with 330,000 images based on the Shapenet 3D objects dataset [19]. They used environmental maps in render software to create shading, reflectance and specular component of objects. The final ground-truth is only based on masked regions of objects without cast shadows information.

*Baslamisli et al. in* [8], follow a similar approach to the previous one. They created an intrinsic dataset of 20,000 images called Shapenet-Intrinsic. Instead of using original shapenet object textures, they used homogeneous color reflectances for each object part to have more reflectance variation and to disassociate the shape from texture. The final dataset is again only based on masked object region enlightened by environmental maps. Example images from their dataset is shown in figure 2.5.

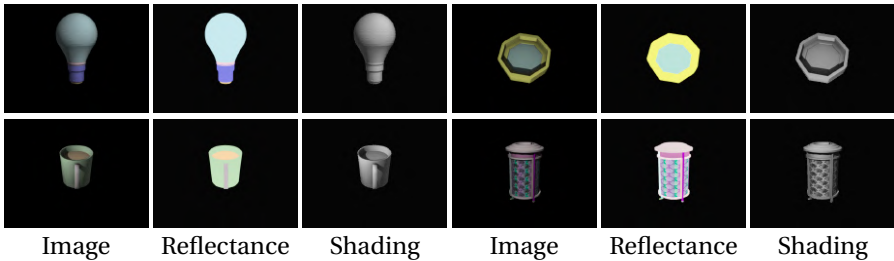


Figure 2.5: Images from Shapenet-Intrinsic dataset

*Baslamisli et al. in* [7] present a new synthetic dataset of 35K images called Natural Environment Dataset (NED). The ground-truth not only provides intrinsic components but also annotations for semantic segmentation purpose. Images are created using sky HDR environmental maps with parallel lighting in the background. These maps introduce different daylight conditions such as clear sky, cloudy, night etc. The whole foreground area has been considerably increased with respect to the previous two mentioned datasets, sky portion areas are masked out in the

ground-truth images. Sample images from NED dataset is shown in figure 2.6.



Figure 2.6: Example ground-truth images from Natural Environment Dataset (NED): left(Original Image), middle (Reflectance), right (Shading)

*CGIntrinsics*[71] is another recent synthetic ground-truth of 20,000 images. They use the 3D indoor objects of SUNCG dataset [114] with 3D textures. It contains images representing complex indoor scenes of objects, combining a mixture of indoor and outdoor illumination. Outdoor illumination sources are also based on HDR environmental maps. Both indoor and outdoor lighting sources are masked out in ground-truth. Figure 2.7 shows sample indoor images(left) with the reflectance image(right) from *CGIntrinsics* dataset .

*InteriorNet* is a very large scale dataset recently presented in [70]. It is formed by 20 millions of images with an extense ground-truth for a variety of applications such as segmentation, object boundary detection, depth map estimation or motion blur removal. They also provide an interactive simulator ViSIM and the rendering software to randomly select objects or change lighting configuration beyond more tools. Although this dataset is not just made for intrinsic decomposition, it provides a good starting point to create a large photo-realistic and physically consistent dataset. In its current state the dataset does not provide intrinsic ground-truth data and the product model is not held.



Figure 2.7: Example images from CGIntrinsic dataset: left(Original Image), right(Reflectance)

To conclude the review of the different available datasets we provide a complete comparison in table 2.1 where we analyze the different datasets according to several properties we have organized in columns referring to: (a) the size or the number of available images with the corresponding ground-truth data; (b) if the dataset fulfills the product model given by Equation 1.9; (c) if the full image can be used in the training, with no mask that reduces the number of samples and meaningful spatial coherence in the training stages; (d) if the ground-truth captures the influence of a diversified background that provoke a large diversity of lighting effects on the object surfaces; (e) if the ground-truth images present cast shadows that add realism to the full scene; (f) if the global lighting is physically consistent with real interaction between all the scene objects, environmental maps usually alter this coherence.

From table 2.1 we can conclude that the majority of available datasets are synthetic, only the first three rows correspond to realistic ones. MIT and MIII are the only realistic datasets that provides full reflectance and shading images, since IIW only give reflectance judgement data for specific pairs. Sintel dataset was not created for this problem but has been one of the most used, since it was the first presenting an enough number of images to train deep architectures, its main problem

## 2.1. Intrinsic Image Datasets

Dataset	Size (# Images)	Model Fulfillment	Training on full Image	Diversified Background	Cast Shadows	Consistent Lighting
MIT ( <i>Grosse et al. [44]</i> )	220	yes <sup>★</sup>	no	no	no	yes
MIII ( <i>Beigpour et al. [10]</i> )	600 <sup>★</sup>	yes	no	no	no	yes
IIW ( <i>Bell et al. [12]</i> )	5230	no	no	yes	yes	yes
Sintel ( <i>Butler et al. [17]</i> )	890	no	yes	yes <sup>‡</sup>	yes	yes
ShapeNet ( <i>Shi et al. [110]</i> )	330K	yes	no	yes	no	no
Shapenet-Intrinsic ( <i>Baslamisli et al. [8]</i> )	20K	yes	no	yes	no	no
<b>Our dataset (SID)</b>	25K	yes	yes	yes	yes	yes
NED [7]	35K	yes	no <sup>†</sup>	yes	yes	no
CGintrinsic ( <i>Li and Snavely[71]</i> )	20K	yes	no <sup>†</sup>	yes	yes	no
InteriorNet ( <i>Li et al.[70]</i> )	20M	no	no <sup>†</sup>	yes	yes	no

Table 2.1: Comparison on current available dataset according to several properties. From left to right we account for: Number of images, ground-truth (GT) perfectly fulfills the physical model, GT is on the full image or only a part, GT is presenting the influence of a diverse background, GT is presenting cast shadows apart from shading, and global image present physically consistent lighting. Meaning of special cases: (★) MIT and MIII datasets generally fulfill product model by including a factor i.e.  $I = \alpha(R \cdot S)$ , but it does not completely hold for all images and have small deviation; (‡) Sintel dataset present diverse backgrounds compared to the rest, but with a strong bias towards specific colors due to high correlation of a video sequences. (†) Training area is large, but still does not cover the full image.

is the high level of correlation between all the images that introduces a dominant color bias.

The three subsequent rows in table 2.1 correspond to Shapenet-based datasets, including our proposal. ShapeNet [110] emerged as a tool to create new larger datasets where synthetic objects are located in multiple different environmental maps. Following this idea, Baslamisli *et al.* [8] used the same approach but using homogeneous reflectance for each object mesh. In both cases the ground-truth is just given by the object area. In the last row we introduce our proposed dataset which is also based on the ShapeNet. Although it is properly introduced in next chapter 3, here we advance some properties that can be compared with the previous ones. Our dataset uses a single reflectance per mesh like in [8], but substituting environmental maps by multiple elements in the scene surrounding the object that

inserts a diversified background, extends the training area to the full image, and adds realism to light effects thanks to shadows and to the physical consistency on rendered light effects. This dataset is presented here as a tunable baseline to easily generate a high diversity of lighting conditions, that can be adapted depending on the task at hand. We explain all the details on how this dataset is built in the next chapter 3.

At the bottom of table 2.1 we have grouped 3 recent datasets that increase the complexity of the scenes, extend the ground-truth to larger areas that can contain cast shadows, but keeping environmental maps in some other parts, which can not be included in the ground-truth and provoke some lack of coherence in the global lighting of the scene. These datasets can be used for more generic applications where a high accuracy in the estimation of light conditions is not required.

## 2.2 Intrinsic Image Estimation Methods

In this section we give a brief overview of single image intrinsic estimation techniques. We have divided these techniques in two sections, first we discuss the classical approaches prior to the arrival of CNNs, which mostly define physical prior on reflectance and shading, Secondly, we discuss deep learning based methods in detail starting from 2015. At the start, these methods were using end-to-end architecture to regress both shading and reflectance from single image. However, recent trend is to incorporate more physical information of traditional methods in deep neural networks.

### 2.2.1 Traditional Methods

Traditional work in this domain was focused on introducing physical priors on reflectance and shading and use optimization approaches to solve the decomposition problem. Earlier solutions in this domain were based on Retinex theory [67] that was based on the hypothesis that shading corresponds to smooth variations while reflectance to abrupt changes in images. Thresholding image gradient can result in two estimations of reflectance and shading . This method was introduced for grayscale images and used later in different works [6, 117, 118, 126]. Funt *et al.*[35] extended this idea for color images by assuming that shading is invariant to chromaticity.

Posterior researchers added more physical constraints to this inverse problem,

like introducing sparsity of reflectance [41, 109], structure of textures [54], new priors on shape and lighting [5], or by adding higher-level color descriptors [104]. In some other works [4, 20, 54, 68] both RGB and depth information were used to predict intrinsic images, they provided promising results, but the lack of larger ground-truths stopped the evaluation of these approaches. Finally, others works focused on illumination varying image sequences from static video cameras [66, 82, 126], in these cases reflectance is constant through all the images. This multiple image-based approaches reduce complexity of intrinsic decomposition, these are very specific applications with special conditions that move away from the single image intrinsic decomposition that is the goal of this thesis.

To conclude the review of the traditional intrinsic image methods we provide a comparison of all of them in table 2.2 where they are classified according to: (a) type of inputs; (b) physical cues they use; (c) intrinsic component they estimate. The majority of methods are based on a single image at the input, but some are based on multiple image video sequences and RGB-Depth images. As discussed earlier, All the initial work solved this problem by assuming that shading varies smoothly in the image as proposed in the Retinex theory [67], subsequent methods added more physical cues based on either on color distributions, texture structures, object shapes or high-level attributes. Finally, we analyze these methods depending on their estimated intrinsic properties. All the methods have focused on the estimation of Reflectance and Shading, with the exception of Barron and Malik [4, 5] that added the estimation of shaped and lighting in the process. In the next section, we review intrinsic image estimation methods after the introduction of convolution neural networks.

### 2.2.2 Deep Learning Based Methods

The broad success of Convolutional Neural Networks (CNN) in computer vision also tooks this community interest on solving this challenging task with supervised networks and by creating big and accurate amounts of data to train. The earlier approaches in this new framework started to appear in 2015. The First work was by Narihira *et al.* [87] that introduced *Direct Intrinsic*s, a two level end-to-end regression network to get shading and reflectance from single image. Their two-stage network was inspired by Eigen *et al.* [31] that predicted depth from a single image. At first level, the network tries to learn a global perspective of the image and at second level it tries to get more fine details from both the input and the first stage output. Second, Zhou *et al.* [136] presented a network to get a relative reflectance between image pixel pairs trained on user annotated data. Therefore, they train

## Chapter 2. Review on Intrinsic image decomposition

Method	Inputs			Physical cues			Outputs		
	Single Image	Multiple Images	Depth Image	Smooth Shading	Color Sparsity	Other	Reflectance & Shading	Shape	Illumination Model
Barrow'78 [6]	✓			✓			✓		
Funt'92 [35]	✓			✓			✓		
Weiss'01 [125]				✓			✓		
Matsushita'04 [82]		✓		✓			✓		
Tappon'05 [118]	✓			✓			✓		
Tappon'06 [117]	✓			✓			✓		
Gehler'11 [41]	✓			✓	✓		✓		
Lee'12 [68]	✓	✓	✓	✓			✓		
Serra'12 [104]	✓			✓	✓	(1)	✓		
Barron'13 [4]	✓		✓	✓	✓	(2)	✓	✓	✓
LShen'13 [109]	✓			✓	✓		✓		
Chen'13 [20]	✓		✓	✓			✓		
Jeon'14 [54]	✓		✓	✓		(3)	✓		
Laffont'15 [66]		✓		✓	✓		✓		
Barron'15 [5]	✓			✓	✓	(2)	✓	✓	✓

Table 2.2: Comparison of tradition intrinsic image decomposition methods based on inputs, physical cues used and outputs. Other physical cues (1) higher level color descriptor (2) shape cues (3) texture structure

on the fact of having similar, larger or smaller relative reflectance difference for each pair [12]. The output of their network is a relative reflectance map for all image pairs, they used the same energy minimization function defined in [12] to get shading and reflectance from this map. And third, Shelhamer *et al.* [107] introduced a different methodology to use both depth and RGB to get intrinsic decomposition. They used a fully convolutional network (FCN) [74] to predict depth from ground-truth image, and they used previous non deep learning methods designed in [4] and [20] to get shading and reflectance from both RGB input and predicted depth. In a similar approach, Kim *et al.* [58] extended the idea to predict depth, shading and reflectance jointly from a deep architecture called joint conditional random field (JCNF) that shares convolutional activation's and layers between all 3 tasks. JCNF learns to predict intrinsic images and depth jointly in a more correlated gradient domain.

Afterwards, Nestmeyer and Gehler [89] showed that prior knowledge on the problem domain can improve the results for methods based on networks. They

introduced a network which learns intrinsic decomposition based on the human judgements given in [12] and proposed a reflective filtering approach, that can be applied at the end of any network, improving prediction efficiency and computational cost. In 2018, Fan *et al.* [32] introduced a multi-stage deep learning architecture. They used a pretrained Direct Intrinsic network [87] to estimate shading and reflectance at a first stage. In parallel they used another network to get a guidance map from the input image edges. In a later stage they used both the first stage intrinsic components and the guidance map to build a more accurate final reflectance. Shading is an element-wise division between the input image the final stage reflectance. This network model is complex and is dataset-dependent.

Few researchers have tried to solve this problem by using Generative Adversarial Networks (GAN) [43]. Lettry *et al.* [69] were the first at it, in a first stage they predicted only shading, and reflectance was considered to be obtained by element-wise division between the input and predicted shading. In a second stage, residual blocks are used to obtain the final shading and reflectance from the first stage predictions. The novelty of this work was to introduce more physical constraint in the loss term and combining data, gradient and adversarial losses together.

Based on a different hypothesis that relies on the use of multiple images of the same scene, Ma *et al.*[79] used a Siamese network (introduced by [16]) to learn a similar reflectance from multi illuminated images. The network is capable of learning to predict shading and reflectance from image pairs using partial supervision i.e. no need of reflectance nor shading ground-truth data. Weak supervision shows significant improvement in their framework. But, at learning time, the network needs image pairs under different lighting which are not always available. On the other hand, Li and Snavely in [72] extended the previous work to train the network based on a sequence of images from the same scene with varying illumination. They collected a dataset of 145 indoor and 50 outdoor scenes and trained a network, that present some bias towards indoor scenes. It shows good performance, but with some blurring effects in the shading predictions, since the network learnt the invariance. All images in the sequence keep the reflectance constant, but a varying shading. This assumption was already posed on previously commented work on intrinsic decomposition [126].

More recently, Liu *et al.* [76] presented a novel method that combines both unsupervised learning with a GAN based architecture for single image intrinsic decomposition, their method is inspired by [50, 51] to learn latent style representations of reflectance and shading from uncorrelated datasets and to apply the content of the input image to get the final intrinsic decomposition.



In spite of unsupervised approaches, there is a common agreement in the fact that CNN-based approaches enhance the need for bigger image datasets. With this aim, Shi *et al.* [110] introduced a large synthetic dataset and an encoder-decoder deep architecture to estimate shading, reflectance and specularity. This is the first deep learning architecture for non lambertian or non diffuse reflectance surfaces, where a specular component is added to the product model. This architecture presents one encoder and three decoders with shared connections. A similar approach was followed by Baslamisli *et al.* [8] that introduced another synthetic dataset and two different deep learning architectures named as *IntrinsicNet* and *RetiNet* to get shading and reflectance from a single image. *IntrinsicNet* is an encoder-decoder end-to-end network to predict shading and reflectance from a uniform architecture. The *RetiNet* has a more complicated scheme to incorporate traditional retinex theory [67] concepts in deep learning approaches. *RetiNet* is a two-stage network, where the first stage learns shading and reflectance gradients, and the second stage combines both input image and first stage gradient information to get the final shading and reflectance. Their claim is the use of more physical constraints on deep networks for a specific application. Similarly, Li and Noah [71] introduced a new synthetic dataset of 20,000 images called CGintrinsic and use a U-Net based network with two decoders to predict shading and reflectance from single image. This network generalizes well on complex scenes, but not providing good results on the MIT dataset. Recently, Baslamisli *et al.* [9] presented a different approach in which they learn initial shading representation by using a traditional retinex-based method [67] and applying a deep learning module to refine the shading image and also to predict the corresponding reflectance image. To train this network, they extended their original ShapeNet based dataset [8] of 20,000 images to 50,000 images.

Finally, recent works on inverse rendering [73, 102, 132] learn reflectance and shading decomposition of an indoor image along with other components such as normal, shape, and lighting environmental map. These methods require all these modalities to train the deep neural network.

To conclude the review of the deep learning based methods we provide a comparison of all these methods in table 2.3 where we organize them based on: (a) type of inputs; (b) type of deep learning network; (c) estimated intrinsic component by each method. From the table we can conclude that most of the research has tackled the problem for single-image input, but more recently, the inverse rendering approaches are extending the prediction to shape, illumination models, maps and other components.

## 2.3. Intrinsic Image Evaluation

In the next section we review how intrinsic image methods are computationally evaluated on existing datasets, and we present our approach to synthesize a diverse datasets to tackle the intrinsic decomposition of light components.

Method	Inputs			Network Type			Outputs		
	Single Image	Multiple Images	Depth Image	Supervised	Unsupervised	GAN	Reflectance & Shading	Specularity	Other
Narihira'15 [87]	✓			✓			✓		
Zhou'15 [136]	✓			✓			✓		
Shellmer'15 [107]	✓		✓	✓			✓		
Kim'16 [58]	✓			✓			✓		(1)
Shi'17 [110]	✓			✓			✓	✓	
Nestmeyer'17 [89]	✓			✓			✓		
Fan'18 [32]	✓			✓			✓		
Lettry'18 [69]	✓			✓		✓	✓		
Ma'18 [79]		✓			✓		✓		
Li'18 [72]		✓		✓			✓		
Baslamisli'18 [8]	✓			✓			✓		
Li'18 [71]	✓			✓			✓		
Yu'19 [132]	✓			✓			✓		(2)(3)
Sengupta'19 [102]	✓			✓			✓		(2)(3)
Li'20 [73]	✓			✓			✓	✓	(1)(2)(3)
Liu'20 [76]	✓			✓	✓	✓	✓		
Baslamisli'20 [9]	✓			✓			✓		
Sial'20 [112]	✓			✓			✓		

Table 2.3: Comparison of deep learning intrinsic decomposition methods based on inputs, type of deep neural network and outputs. Other outputs (1) Shape (2) Normal Map (3) Illumination model

## 2.3 Intrinsic Image Evaluation

Qualitative and quantitative evaluation provides a way to compare the performance of different algorithms and it's considered as an essential step in different engineering fields. Earlier methods relied only on visual analysis as there was no ground-truth available for very long period of time. This visual analysis mostly relied on how much shading and shadows artifacts are present in the reflectance image.

After introduction of MIT intrinsic dataset[44], There are three most used metrics for Intrinsic image evaluation and comparison. Mean-squared error (MSE) and

the local mean-squared error (LMSE) are more data-related metrics and structural dissimilarity index (DSSIM) is more a perceptual metric. In the formulation below,  $x$  denotes a ground-truth image and  $\bar{x}$  denotes predicted image.

*Mean-square error (MSE)* measures the mean square difference between estimated and ground-truth pixels in reflectance, shading or the product of these. Similar to [20, 44, 87], we also used scale-invariant metric to remove absolute brightness effect:

$$MSE(x, \bar{x}) = \sum_{i=1}^N \frac{\|x_i - \bar{\alpha} \bar{x}_i\|^2}{N}, \quad (2.1)$$

$x$  can be reflectance  $R$ , shading  $S$  or their product,  $R \cdot S$ , and  $N$  is the total number of pixels in the evaluated estimation. The factor  $\bar{\alpha} = \operatorname{argmin}_{\alpha} \|x_i - \alpha \bar{x}_i\|^2$  adjusts the absolute brightness between estimation and ground-truth to minimize the error and have scale invariance in metric.

*Local mean-squared error (LMSE)* is used to compute scale-invariant MSE on overlapping windows of 10% of the image size. If image is not square, the larger dimension is used to compute windows size.

*Structural dissimilarity index (DSSIM)* is a dissimilarity version of the structural similarity index (SSIM). Differently, from the other metrics that measure absolute error between pixel values, SSIM is based on structural differences. It is motivated by human perception for structural difference.

$$SSIM(x, \bar{x}) = \frac{(2\mu\bar{\mu} + c_1)(2\sigma_{x\bar{x}} + c_2)}{(\mu^2 + \bar{\mu}^2 + c_1)(\sigma^2 + \bar{\sigma}^2 + c_2)}, \quad (2.2)$$

where  $\mu$ ,  $\sigma^2$  and  $\sigma_{x\bar{x}}$  are mean, variance and co-variance respectively;  $c_1$  and  $c_2$  are used to control zero approaching denominator. DSSIM is defined as

$$DSSIM(x, \bar{x}) = \frac{1 - SSIM(x, \bar{x})}{2}. \quad (2.3)$$

## 2.4 Conclusion

In this chapter, we gave an overview of existing datasets and techniques for intrinsic image decomposition. Firstly, we briefly discuss each of the available intrinsic

image datasets and provided a complete comparison in table 2.1 where we analyze different datasets according to several properties. The emergence of deep learning based models makes it necessary to build large intrinsic image datasets. However, the complexity and challenges in building such a large realistic dataset forced most researchers including us to build synthetic datasets with better and realistic illumination effects. Secondly, we revised traditional and deep learning based models that have been developed to predict intrinsic components, we also provided a comparison of traditional and deep learning based models according to different parameters in table 2.2 and 2.3. Here we noticed that most traditional and deep learning based methods still focuses on predicting reflectance and shading properties only using a single image but this trend is gradually changing to include additional components of specularities, shape, normal map, and illumination model. Finally, we discussed the most commonly used evaluation matrices for intrinsic decomposition in section 2.3

In the next chapter, we will present our approach to synthesize a diverse dataset to tackle the intrinsic decomposition, and jointly with it we will propose a new deep learning architecture to predict both shading and reflectance in parallel.



# 3 Reflectance and Shading Estimation

This chapter focuses on getting basic intrinsic image properties of reflectance and shading from a single image. In the first half of this chapter, we briefly describe the challenges associated with ground-truth generation for the intrinsic image decomposition task and four potential approaches: (a) Data augmentation; (b) Physical acquisition of intrinsic properties; (c) Construction of combined synthetic-real datasets; (d) Synthetic ground-truth generation. In the second half of this chapter, we propose a deep neural network that uses the physical constraints of the intrinsic model to estimate reflectance and shading components. We show the results on a set of experiments that evaluate the performance of the proposed architecture trained on our synthetic surreal dataset and tested and compared with other methods and datasets. We prove a promising generalization power of our SID dataset.

## 3.1 Introduction

The basic intrinsic image problem is to split a given single image in its shading and reflectance components, where shading captures the effects caused by interactions between scene illumination, camera position and surface geometry, and reflectance image captures the chromatic properties of the surface object. Both concepts were introduced in more depth in chapter 1.

Current research in the estimation of these components is dominated by data driven methods. End-to-end deep learning architectures have shown promising

results but they rely on having large datasets to be trained on, which is a clear obstacle considering that building datasets for this complex physical problem requires complete control of lights and careful manipulation of objects and its material properties that is only possible in accurate laboratory conditions. The complexity of making accurate ground-truths for this task is supported by the fact that there are only two physically captured datasets up to now; the MIT Intrinsic [44] and the MIII dataset [10], both having only 20 and 5 different scenes respectively which are beyond from being used to train any deep learning model. We have spent some efforts in getting such large datasets by overcoming aforementioned problems. We have followed three different approaches to face this problem that we summarize in what follows.

The first and easiest way to tackle the problem was the use of a color-based data augmentation technique that extends the training data by increasing the variability of chromaticity in an opponent space. The lack of data is partially solved with this data augmentation, but geometric changes are not augmented with this technique.

The second approach, we worked on was a complete pipeline to build datasets based on registering synthetic with acquired images to be able to automatize the process of building the dataset ground-truth. This is an ambitious objective that made us to face a list of complex problems like: (a) setting a platform with pan-tilt movement where to create a diverse geometry in the scenes; (b) calibrating light and camera physical parameters; (c) representing the physical world in a Computer Graphics render engine (Blender [23]); (d) registering acquired with synthetic images in order to synthesize some intrinsic components and derive the rest using the intrinsic product model.

Apart from the complexity of setting the full system, we found that mimicking partial intrinsic versions of the real-world scenes in the synthetic world is a highly complicated task, we were able to achieve high accuracy in duplicating the real-world geometric setting in the synthetic world but making accurate representation of photometric environment in computer graphics is a more difficult task and it still remains an open problem for future research. We succeeded in using this hybrid ground-truth creation setup in a specific application, we created a Doc3Dshade dataset that automatically acquired a large amount of illumination effects on white warped paper surfaces as intrinsic shading, jointly with intrinsic depth and it was combined with synthetic reflectance components. A large ground-truth was created in this combined pipeline to create a dataset of document images to improve the performance of text recognition systems. More details on this can be found in chapter 5.

The third approach was handled in parallel with trying to overcome the challenges associated with the complexity of the second approach, we worked on building completely synthetic datasets. We created the SID family for *Surreal Intrinsic Dataset*, this datasets were created to achieve a lot of variations of shading and reflectance effects in an automatic way pursuing photometric realism avoiding the use of environmental maps and using a large variety of objects and backgrounds.

## 3.2 Ground-truth Generation

In this section we explain the details of the three approaches we have worked on in this thesis for ground-truth generation. Additionally, we also explain the difficulties of the real acquisition of intrinsic components, that are in the basis of our second approach.

### 3.2.1 Data Augmentation

Deep convolutional architectures have become a flexible tool to solve problems that has provoked methodological changes in the design of computer vision solutions. One of these changes is that important research interests have shifted from the design of the solutions towards the setting of adequate experimental setups to find the best hyper-parameters to reach best performances. As we mentioned before, data augmentation in the training stage is one of these methodological aspects that attracts some attention.

We propose a data augmentation technique that seeks to extend the color diversity of datasets. It is based on the idea that for any given image we can apply a chromaticity rotation without affecting intensity. This is a transformation that in the intrinsic model affects reflectance but does not affect shading. This chromatic change should be equally applied to the ground truth reflectance and to the original image to hold the intrinsic decomposition property. In this way we achieve an extension of the image dataset from a photometric point of view instead of the most common geometric extensions. This augmentation increases the dataset color variability and we prove it increases the generalization capabilities of the network architecture.

To this end we use an opponent-like (CieLab-like) transformation where intensity and chromaticity are separated. Afterwards, we apply a rotation on the plane formed by the red-green and blue-yellow axes. The transform to this opponent



space is a linear transform on the RGB color space given by the following equations:

$$\begin{aligned} O_1 &= (R + G + B - 1.5)/1.5, \\ O_2 &= (R - G), \\ O_3 &= (R + G - 2B)/2. \end{aligned} \tag{3.1}$$

It is based on the one proposed by Platanoids *et al.* [96] but normalizing and shifting the three axes within the range  $[-1, 1]$ . This space was conceived to achieve certain physiological inspiration on uncalibrated RGB, and has provided interesting results in computer vision.

Augmentation can be reached by random rotations which are computed on the  $O_2$ - $O_3$  plane, followed by a stretching transform, denoted with the function  $S$ , that keeps the original contrast of the image which could be reduced by the new range after the rotation

$$\begin{aligned} O_2^\theta &= S(O_2 \cos(\theta) + O_3 \sin(\theta)), \\ O_3^\theta &= S(O_3 \cos(\theta) - O_2 \sin(\theta)), \\ S(O_i) &= (O_i - \min(O_i)) \frac{\max(O_i^\theta) - \min(O_i^\theta)}{\max O_i - \min O_i} + \min(O_i^\theta). \end{aligned} \tag{3.2}$$

We can see an example of the effects of this chromaticity rotation in figure 3.1, where we also can see how the transformation holds the hypothesis of preserving the shading while changing reflectance.

The proposed data augmentation technique improved the performance of the reflectance estimation method [111] on all the metrics presented in chapter 2. We reduced mean-square error (MSE) from 0.025 to 0.0231, local mean-squared error (LMSE) from 0.016 to 0.013, and Structural dissimilarity (DSSIM) from 0.26 to 0.24 on the Sintel scene split dataset [17].

### 3.2.2 Real Datasets

The most common procedure to create realistic ground-truths for intrinsic image decomposition is using a gray-paint spray. To be specific, first, a scene with a single or multiple objects is captured under all lighting configurations, and then these objects are painted with gray spray and placed at the exact same location and captured again under the same illuminations. Gray painted scene image is considered as the shading ground-truth, and the reflectance ground-truth is obtained by dividing the

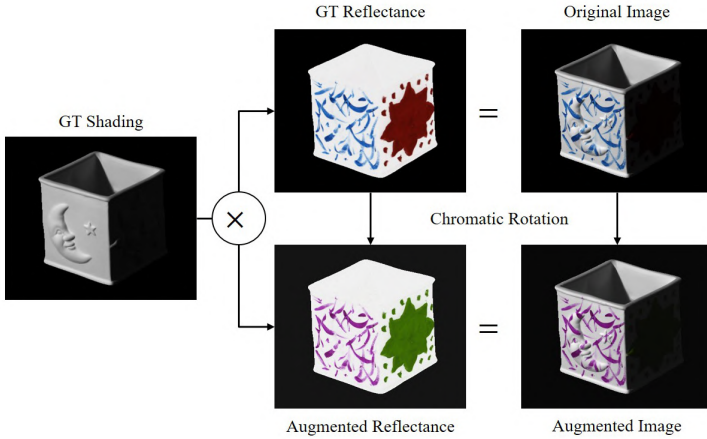


Figure 3.1: Example of chromaticity rotation for color-based augmentation.

image of the non-painted object by the image of the painted one. Following the intrinsic physical model of equation 1.9.

$$R(x, y) = I(x, y) / S(x, y) \quad (3.3)$$

This procedure was firstly applied in the most popular and commonly used MIT intrinsic dataset by Grosse *et al.* [44]. It has 20 objects, each one captured under 11 different lighting conditions, and for every light configuration the ground-truth provides: (1) the original image, (2) a diffuse version of the original image captured with a polarizing filter to remove specularities, (3) a shading image that is the acquisition of the gray-painted object, (4) a binary mask of the object profile, (5) a reflectance image computed by division, and (6) a specular image computed by subtracting the diffuse image from the original image.

Figure 3.2 shows 7 out of 20 objects in the MIT Intrinsic dataset [44]. This dataset was the first satisfactory proposal for an intrinsic image dataset and it has been widely used ever since for the quantitative and qualitative evaluation of intrinsic image estimation methods.

Beighpour *et al.* in [10] followed a similar approach to create the MIII dataset which in principle is an extension of the MIT Intrinsic for a multi view multi illumi-

nant scenario. The dataset has 5 different scenes and each scene is captured under 20 different illumination conditions with 6 different cameras views, giving the total of 600 images. Inspired by MIT intrinsic dataset [44], each object in the scene is painted in white and captured again to get the ground-truth reflectance again by dividing the color image with the painted image. They also provided raw depth and 3D point cloud information. Figure 3.3 shows five scenes of the MIII dataset captured by one of six cameras.



Figure 3.2: Objects in MIT Intrinsic dataset



Figure 3.3: Five scenes of MIII dataset

Although, creating a dataset in this manner does provide satisfactory ground-truth information with perfect realism but it has certain disadvantages. Firstly, the method to build the shading image by painting objects in white and placing them at the same location makes the process to be tiresome and slow, which explains the small number of objects in the scenes and the limited size of the ground-truth in both MIT intrinsic [44] and MIII dataset [10]. Secondly, to reduce undesirable illumination effects, such as inter-reflections and highlights, most of the objects used in these dataset have essentially convex shapes with diffuse materials. Finally, the usage of matte gray paint also reduces important illumination effects in the scene.

To resolve these shortcomings, in the next section, we make preliminary proposals, we could not fully achieve, as a novel hybrid procedure to create combined

synthetic-real intrinsic image dataset.

### 3.2.3 Combined Synthetic-Real Dataset

Since deep convolutional networks are solving computer vision problems, the need for large image datasets is a must. Physically-based large image ground-truths for intrinsic image decomposition are difficult to build, and this has become one of the main drawbacks to progress in this problem. In this section, we propose a method to solve this problem by registering acquired images with synthetic images that can generate some of the intrinsic components and simplify the tough acquisition task and the corresponding ground-truth generation.

Our hypothesis is that if we can replicate the real-world geometric and photometer setting in the computer graphics world with enough accuracy, we can overcome shortcomings of real dataset creation and build a huge dataset with enough variation of lighting, objects, materials, and camera views that would be better suited for current deep learning based approaches.

#### Lab Setup

To build this combined synthetic-real dataset, we created a lab setup with fixed cameras and lights position and a rotatory platform to hold the objects. We covered the background walls with black curtains to remove undesirable illumination effects of ambient lights and inter-reflections.

Our platform has  $19 \times 19 = 361$  holes to hold the objects on top. To avoid displacement of objects during platform movement and image acquisition, we opted for hexagonal shape holes on the platform surface. The platform movement is controlled by a computer and it has 2 degrees freedom i.e. that it can move both in pan and tilt directions. The platform can move  $[0, 360]$  in pan direction and  $[-45, 45]$  degrees in tilt direction. We have also added sensors in the platform to go to the global 0 or platform home position.

To create multi illumination detest with color lighting, we placed  $3 \times 3$  grid of led bulbs at  $100m$  height from the platform. All 9 bulbs are directed towards the center of the platform. We used Philips hue color lights with a bridge to control light parameters. The Philips bridge is connected to the computer and has a wireless connection with each light to control their properties. We can program to switch on/off each light, change the brightness and color of each light. We also installed

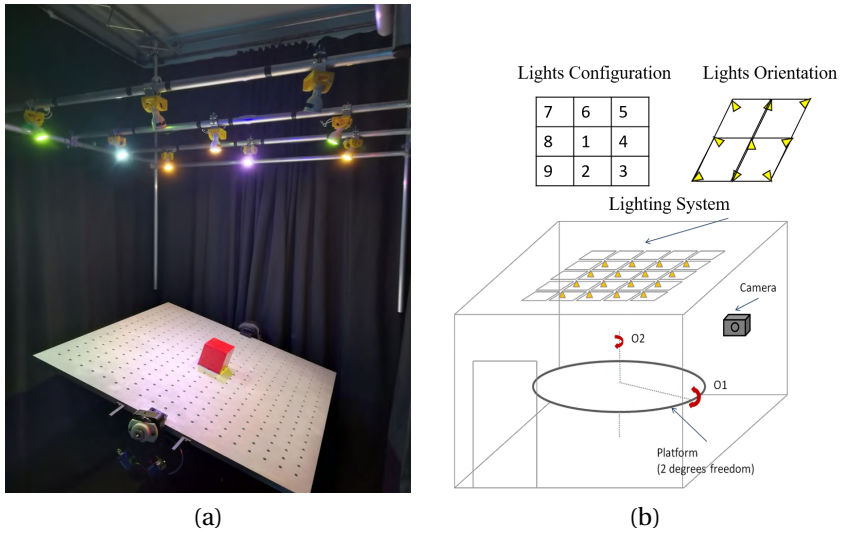


Figure 3.4: Here we demonstrate our lab setup (a) Platform image in which a red cube is placed at the center of the platform, image is illuminated by 9 light bulbs from the top and platform is tilted towards one side (b) graphical representation of our lab setup, platform has 2 degrees of freedom and it can move in both pan and tilt directions, light configuration and light orientation are shown on top, all our lights are pointing towards center of the scene.

polarize filters with motors on each led bulb, these motors are connected to Arduino to capture images with/without specularities. The position and direction of all led bulbs remained fixed and untouched during image acquisition.

Our initial camera setup had three RGB cameras namely Nikon D5200, Samsung Galaxy S7 mobile phone, and Logitech C920 webcam placed closer to each other. All three cameras have very different sensor responses. The position and direction of all three cameras are fixed and they are directed towards the center of the scene and capture images from the front-top. To create reliable detests, we disabled auto white balance and manually set different camera parameters, and kept them fix. Nikon D5200 and Samsung Galaxy S7 cameras can capture images both in RAW and JPEG format but the webcam can only capture images in JPEG format. Figure 3.4.(a) shows the image of our lab setup in which a red cube is placed at the center of the tilted platform and the image is illuminated by 9 Led light bulbs from the top. 3.4.(b) illustrate the graphical representation of our lab setup with light configuration and light orientation on top.

To create Doc3DShade dataset, we added Intel RealSense RGB Depth Camera D435 camera on top of the platform, more detail on this is provided in section 5.1.2

### Acquisition Setup

First step in making dataset is to apply calibration between physical and rendering world. Following calibration steps are performed to get parameters of lights, cameras and platform in Blender [23].

*Step 1: Camera calibration.* We used Matlab camera calibration toolbox [129] with a standard pinhole camera model to estimate intrinsic and extrinsic parameters of cameras. This estimation is based on a  $20 \times 21$  checkerboard pattern with each square size of  $40mm$ . In this calibration step, we get images by rotating and tilting the platform at multiple positions. Cameras are horizontally aligned with a small distance between them and their positions kept fixed for all later acquisitions.

Camera parameters are generally represented by a  $4 \times 3$  matrix, we denote as  $\mathbf{P}$  that maps 3D world points to image plane points and vice versa. This matrix comprehend both the camera extrinsic and intrinsic properties, the extrinsic matrix represents camera position in 3D world, and the intrinsic represents optical properties of the camera including the focal length  $(f_x, f_y)$ , the principal point  $(c_x, c_y)$  of the camera and the skew coefficient  $(s)$ , in this

way

$$\mathbf{P} = \begin{bmatrix} \mathbf{R} \\ \mathbf{t} \end{bmatrix} \mathbf{K} \quad (3.4)$$

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & 0 \\ s & f_y & 0 \\ c_x & c_y & 1 \end{bmatrix} \quad (3.5)$$

where,  $\mathbf{K}$  is  $3 \times 3$  intrinsic matrix and  $\begin{bmatrix} \mathbf{R} \\ \mathbf{t} \end{bmatrix}$  is extrinsic matrix formed by a  $3 \times 3$  rotation matrix  $\mathbf{R}$ , and  $3 \times 1$  translation matrix,  $\mathbf{t}$ . The extrinsic matrix maps any point from world coordinate system  $[X, Y, Z]$  to the camera coordinate system  $[X_c, Y_c, Z_c]$  and the intrinsic parameters are used to transform camera points to the image plane  $[x, y]$ .

*Step 2: Estimating light positioning.* To obtain the light direction and position in the real-world, we used the ideas proposed by [101] which is based on surface reflection in the DRM model [106] (see section 1.1) for the specular ball. The surface reflection is a mirror like reflection in which the angle of incident light is equal to the angle of outgoing light at the specular highlight on a spherical ball (see 3.5(a)). We placed the specular ball of  $150mm$  at the center of the platform (3.5(b)) and illuminated it separately by all lights and applied this algorithm to get a rough estimation of each light position and direction. The algorithm forms a vector ( $\mathbf{V}_i$ ) from the camera center ( $O$ ) through highlighted pixel point in the image and locates its intersection on a sphere with highlight. By using the surface reflection property we can recover light vector ( $\mathbf{V}_r$ ) as:

$$\mathbf{V}_r = \mathbf{V}_i - (2\mathbf{N} \cdot \mathbf{V}_i)\mathbf{N} \quad (3.6)$$

where

$$\mathbf{V}_i = \frac{\mathbf{K}^{-1}x}{|\mathbf{K}^{-1}\tilde{x}|} \quad (3.7)$$

is the viewing vector constructed from highlighted pixel point  $\tilde{x}$  in the image and the camera intrinsic matrix  $\mathbf{K}$  from step 1. This algorithm provides only light orientations, to get light position, we used the fact that lights are placed at  $100m$  height from the platform.

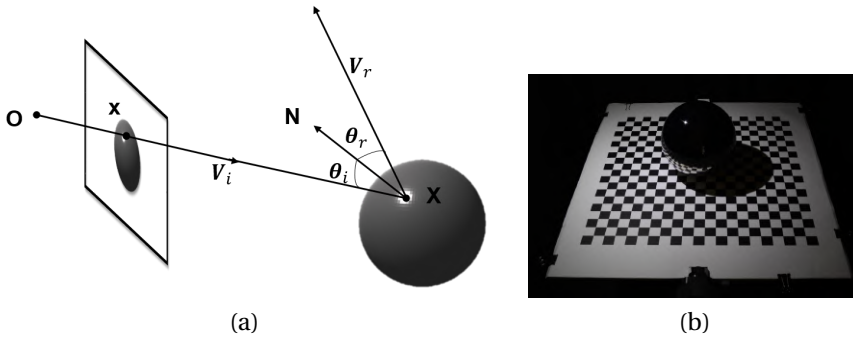


Figure 3.5: (a) The image illustrates that the angle of incident light is equal to the angle of outgoing light at the specular highlight on a spherical ball (b) our Light calibration setup in which we placed 150mm ball in the center of the platform

*Step 3: Setting synthetic world.* We made a replica of our lab setup together with platform, lights, and camera in the computer graphics world (Blender [23]) and transform MATLAB [81] parameters to Blender parameters. In our setup, we assumed origin to be the center of the platform, so we changed camera and light position from MATLAB to our world by changing axis and applying shift to center (figure 3.6).

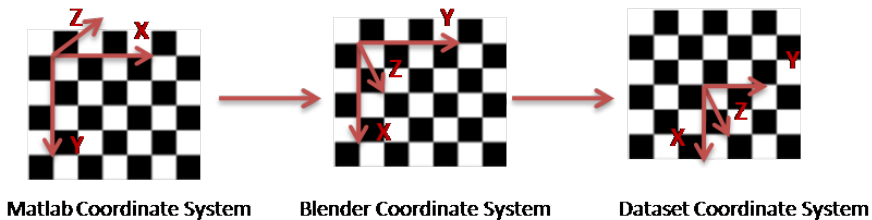


Figure 3.6: Coordinate Systems in MATLAB [81], Blender [23] and our Dataset, respectively from left to right.

After setting synthetic world, we defined errors maps to evaluate camera and light calibrations. To check the accuracy of the camera calibration, we placed a similar checkerboard pattern in Blender and calculate the difference between checkerboard points in both the real and the synthetic world, and interpolated the error to the checkerboard area. Figure 3.7 shows evaluation



with error calculation for camera calibration. Whereas, in the case of light calibration, we covered the platform with a simple white plane in both the environments and take the absolute difference between white patches as an error measure to validate its accurateness. Figure 3.8 shows evaluation with error calculation for the light calibrations, we take the absolute difference between the intensity of the real and synthetic image central patch marked with blue rectangles.

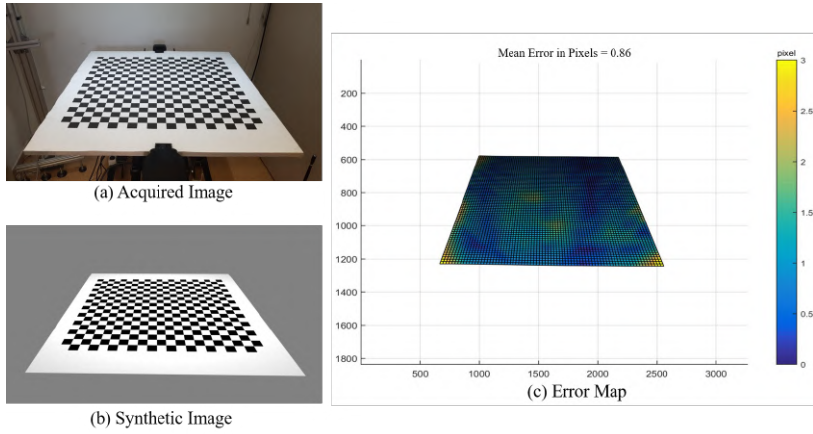


Figure 3.7: Camera Calibration Evaluation: (a) Acquired image from real-world (b) Synthetic image with parameters from camera calibration (c) Interpolated error map based on the difference between checkerboard points in real and synthetic image

#### *Step 4: Refinement of camera position.*

To adjust camera parameters(position and orientation) in Blender, we placed color pattern (see figure 3.9.(a) bottom left) in the center of the platform in real and Blender world and used the distance between four color centroids in the real and synthetic world as an error metric in simulated annealing (SA) [61] to refine camera parameters in Blender. Figure 3.9 demonstrates the results with and without using SA, the bottom row images (d,e) shows the blended images between the real and synthetic world before and after using SA respectively with the cropped color pattern at the bottom left, these images shows that SA can help to refine camera parameters in Blender world to better align with the real-world.

### 3.2. Ground-truth Generation

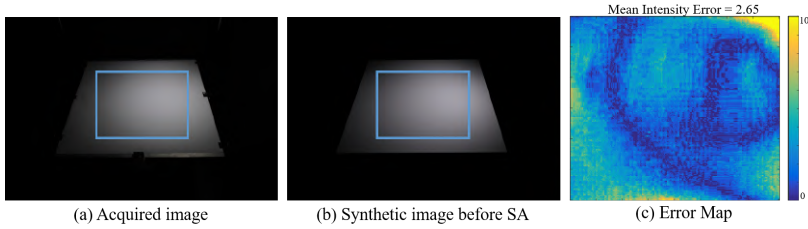


Figure 3.8: Camera Calibration Evaluation: (a) Acquired image from real-world (b) Synthetic image with parameters from light calibration (c) Error map as the absolute intensity difference between the real and the synthetic image central (marked with blue rectangles in (a) and (b))

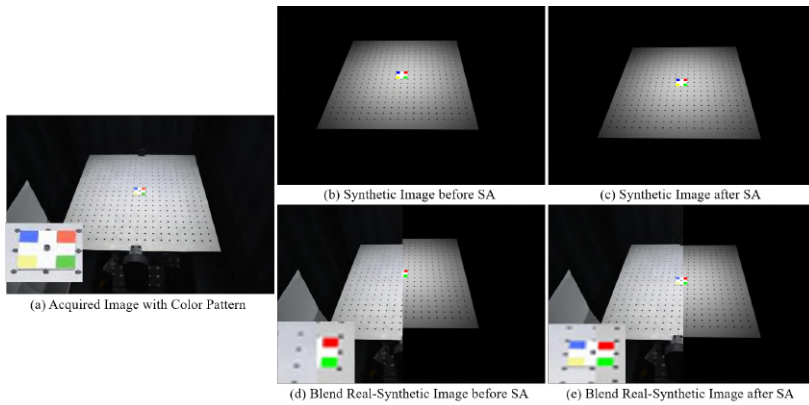


Figure 3.9: Refinement of camera position: (a) Acquired image from real-world (b) Synthetic image with parameters from camera calibration (step 2) before simulated annealing (SA) (c) Synthetic image with parameters from camera calibration (step 2) refined with simulated annealing (SA) (d) A blended image of (a) and (b) with a color pattern at the bottom left. (e) A Blended image of (a) and (c) with a color pattern at the bottom left.

*Step 5: Refinement of light position.* To adjust the Blender parameters(intensity, cone size, and shadow decay) that fit the gaussian that renders the light beam, we covered the platform in white and used its difference in Blender and real-world as an error in the simulated annealing (SA) to optimize light parameters. The overall effect of optimizing the light parameters in Blender

with SA can be seen in figure 3.10 with the mean intensity difference between real and synthetic images central regions marked with blue rectangles, SA helped to reduce intensity difference from 2.65 to 1.61. The bottom row images show a white sphere in the real and synthetic world after SA, the overall resemblance in shading and specular highlight in both world spheres shows the effectiveness of the proposed calibration pipeline.

*Step 6: Estimation of sensor correction.* To correct a full 3x3 camera color correction to the Blender camera, we applied linear minimization using a Macbeth chart.

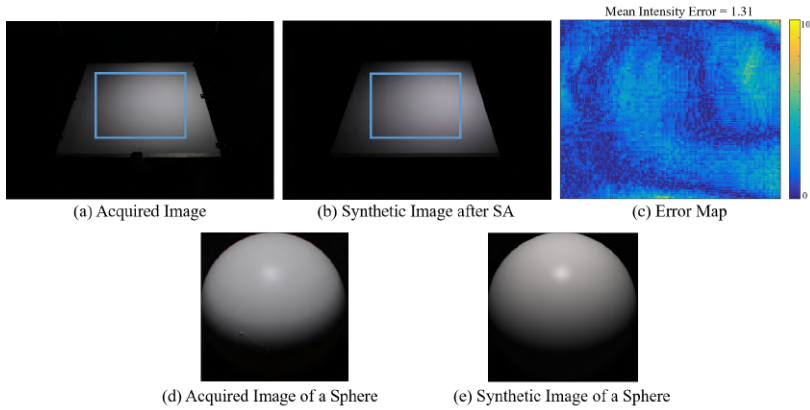


Figure 3.10: Refinement of light position: (a) Acquired image from real-world (b) Synthetic image with parameters from light calibration (step 2) after simulated annealing (SA) (c) Error map as the absolute intensity difference between the real and the synthetic image central (marked with blue rectangles in first two images) (d) Acquired image of a sphere (e) Synthetic image of a sphere

#### Dataset Acquisition

In this section, we explore different ways to create an intrinsic image dataset with their challenges in our lab setup. The first and most straightforward approach to create the dataset would be to use MIT intrinsic [44] gray shading technique in which all ground-truth comes from real data. To validate this technique with our setup, we first placed a cube wrapped in white on our platform and captured shading image, then we painted the cube and placed it roughly at the same place to capture a color image, ground-truth reflectance image is generated by dividing color

image with the white shading image. To remove the effect of small displacement, we register both images. The ground-truth images for this experiment are shown in figure 3.11. As discussed earlier, The main challenge for creating GT intrinsic images in this way is to paint the objects and capture them twice, and to minimize the movement effect while removing and putting objects at the same location. These challenges reduces the overall scale of the dataset.

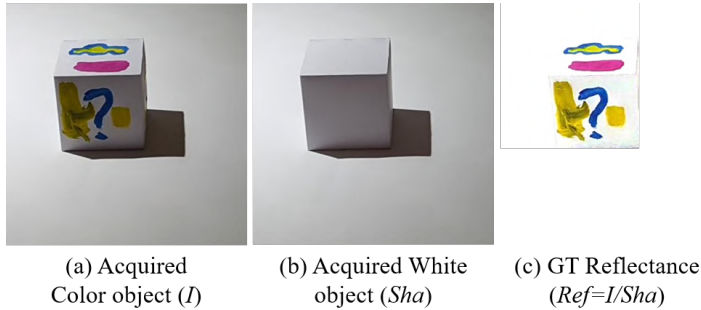


Figure 3.11: Example of GT data with our lab setup using MIT intrinsic[44] dataset creation technique: (a) Image captured in real (b) white Shading is captured in real (c) GT Reflectance is created by dividing color image with white shading

Once both the real and the synthetic world are registered, reflectance and shading components can be rendered in the synthetic world and the missing GT component can be generated by following intrinsic image model. In the second approach, we created a Doc3DShade dataset to remove illumination effects from document images and to improve the performance of automatic document content processing algorithms. This dataset is created in a combined manner in which we capture the 3D shape and illuminant shading of warped papers with different materials  $MS$  under real-world lighting. The captured image has both material  $M$  and shading  $S$  components. Later, we render reflectance textures  $T$  using the captured 3D mesh in Blender [23] and multiplied it with the captured image to create the ground-truth image  $I = MS.T$  (see 3.12). To captured aligned shape and shading images together we added Intel RealSense RGB Depth CameraD435 camera on top of the platform.

Doc3DShade is the extension of a completely synthetic Doc3D document dataset [26]. In comparison to its completely synthetic pair, it captures physically accurate and realistic shading under complex illumination conditions, which is impossible to obtain with rendering engine only. Doc3DShade also contains various paper materials with different reflectance properties whereas a synthetically rendered

dataset like Doc3D uses purely diffuse material to render images. Doc3DShade is thus physically more accurate and can be used to model scene illumination in a physically grounded way. The limitation of creating dataset in this manner is the inexpensive sensor resolution to capture highly accurate 3D shape and alignment of 2D texture coordinates with 3D shape to create GT reflectance images. In the case of document images, overall shape, size and alignment are always fix and known which is not true for more complicated 3D shape objects. We introduced the Doc3DShade dataset in section 5.1.2.

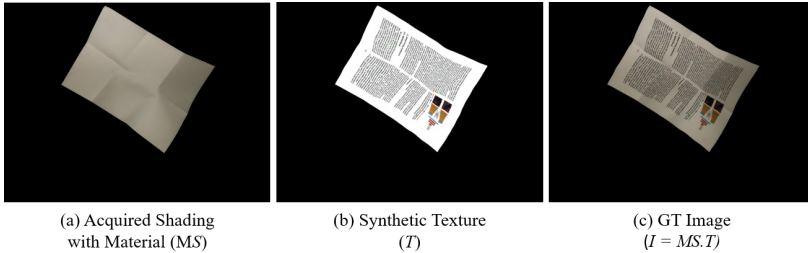


Figure 3.12: Example GT images from DoC3DShade Dataset: (a) Shading with document Material is captured in real (b) reflectance texture is rendered in the synthetic world (c) and GT image is generated by multiplying acquired with synthetic.

Our third proposal to create ground-truth data for this task is more similar to MIT intrinsic[44] dataset creation pipeline. We propose to overcome the complexity of the ground-truth acquisition in their real conditions by using rendering tools. Therefore, we can estimate the reflectance and the shading components of natural objects by using their 3D scanned information to synthesize the Shading of the corresponding white object. We first register real acquired images with their corresponding synthetic versions in terms of their geometric and photometric setting by using the 6 steps defined in the previous section. After registration of two worlds, we captured the color image in real and rendered the shading image in white by using the 3D model of objects/scene and created GT reflectance by dividing real image with rendered shading. The proposed methodology allows us to produce a simplistic initial dataset where 3D models of primary objects can be easily constructed (see figure 3.13). In the case of more complex 3D shape, we tried different Multi-view Stereo with structure from motion (SFM) algorithms [18, 36, 127] to reconstruct 3D shape of objects. These algorithms requires a set of photographs of an object or a scene from the different camera positions and we can easily capture these images by rotation our platform. The general pipeline of these SFM algorithms is: (a) detect and match 2D features between images (b) get a point spares reconstruction by

### 3.2. Ground-truth Generation

using initial matched points (c) dense reconstruction to get final 3D shape. The major challenge in building this combined synthetic-real dataset is that: (a) accuracy of these 3D reconstructed models was not sufficient enough for our dataset creation pipeline, we propose to improve this accuracy in future by using a more sophisticated 3D scanner to provide accurate 3D shape. (b) we were able to achieve high accuracy in duplicating real-world geometric settings in the synthetic world but making an accurate representation of the lighting environment in computer graphics is a more difficult task and it remains an open problem for future research.

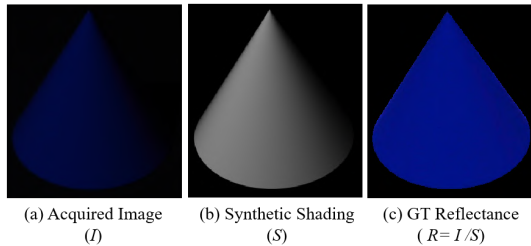


Figure 3.13: Example GT images from our combined synthetic-real Dataset: (a) GT Image is captured in real (b) shading image is rendered in the synthetic world (c) GT Reflectance is generated by dividing acquired image with rendered shading.

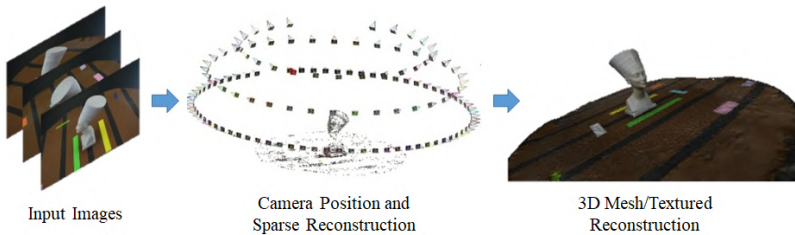


Figure 3.14: Multi-view Stereo: (left) set of images from different viewpoint, (middle) sparse reconstruction by using structure from motion algorithms (right) Dense reconstruction from initial sparse reconstruction. We used Visual SFM [36, 127] to get 3D reconstitution in this image.

Our final proposal to create a large intrinsic image dataset is based on the calibrated photometric stereo algorithm [98]. This algorithm requires a set of images under different illumination conditions with calibrated lights and cameras. We captured 9 images by switching on each light and applied this algorithm to get

an estimation of reflectance, normal, and shape (see figure 3.15). The estimated reflectance is not accurate to be called as ground-truth for an intrinsic dataset. In the future, we will further investigate these photometric stereo algorithms and combine them with our combined synthetic-real dataset creation pipeline to improve the quality of estimated reflectance, shape, and normal.

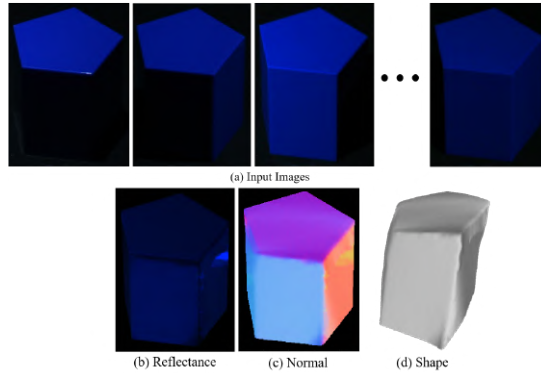


Figure 3.15: Photometric stereo: (a) input images with calibrated lights and camera (b-d) reflectance, shading and shape estimation with [98]

### 3.2.4 Synthetic Dataset : SID1

To overcome the challenges of real and combined synthetic-real datasets, We proposed a synthetic dataset that can be as large as needed, presents realistic light conditions and which can be easily adapted to different tasks e.g. estimation of light components(chapter 4 and single image relighting 5). This synthetic dataset is based on Shapenet objects [19] without textures like in [8], although they could be added. However the main difference is that we improve realism of light effects and training capabilities. We pursue a dataset with the following properties: (a) existence of cast shadows and global light consistency; (b) keeping the influence of a varying surround; and (c) taking advantage of using the full image information for training.

To achieve the previous advantages we substitute environmental maps by multi-sided rooms with highly variable reflectances on the walls. Although the current dataset uses multi-sided flat walls, the shape of the rooms can be extended to multiple wall shapes, such as cylindrical or warped. In this dataset, we focus on a

simple version to start testing the approach on intrinsic decomposition. The main property of environmental maps is to get diverse lighting conditions, since they allow to assume a different light source at every point of the background, but the drawback of these maps is that they are unable to cast shadows and introduce some physical inconsistencies in the light interactions of the scene. A synthetic scene with walls covered by textured patterns and some point light sources at different random positions in the room can be a useful background to simulate a large number of scene images with the three proposed properties.

Our dataset has 25,000 ground-truth images with intrinsic data for the whole image pixels. We used the open source Blender rendering engine to generate images, which can be used with multiple GPU's making rendering much faster. Below we list the key features of our dataset:

**Objects:** We used 12,500 3D objects from Shapenet[19], randomly selected from several object categories such as bus, car, chair, sofa, airplane, pots, electronics etc. Similarly to [8], we also observed that object textures affected Blender shading, resulting in a wrong reflectance and shading decomposition in the generated ground-truth, to solve this issue, we also used a diffuse bidirectional scattering distribution function (BSDF) with random color and roughness values for each mesh texture in object. Roughness parameter controls how much light is reflected back from each object surface. Our dataset presents significant shading and reflectance variation across different objects and its surfaces. Objects are positioned in the image center.

**Backgrounds:** To have diverse backgrounds of shading and reflectance, we defined 4 to 6 sided rooms (see figure 3.16.(c)), where the number of flat walls was randomly selected for each image. These rooms provide a wide range of different geometric configurations with diverse light reflection conditions on the objects. Walls were colored with 50 homogeneous color and 200 variable set of textured patterns. Textured images were carefully selected from Corel dataset [24], we choose the subset *TexturePattern*, formed by fabrics, and *Marbles*, with the aim to ensure they were flat surfaces with no shading effects projected on the image texture pattern (see figure 3.16.(d)). In the current version of the dataset we have used a unique color or texture for all the walls in the scene, in order not to increase too much the variability of this first version, but this can be introduced as an additional parameter to increase the variability of the dataset.

**Generation setup:** To generate random scenes we locate the object at the center



of the scene, where we also put the centroid of the volume formed by the walls, fixing the distance between the object and the walls within a range that ensures that all objects fit in the room. The camera is placed at a fixed distance from the center of the scene, the position is randomly selected on the defined semi-spherical surface. The camera image plane is orthogonal to the line from the center. Each object is captured twice, by using two camera positions separated by 180 degrees in the horizontal pan axis of the semisphere, in this way, we can get two different views of the same object. To set the scene lighting we put 4 white light sources with fixed location and orientation, we just randomly changed the intensity, that is what most affects the intrinsic shading. Two of the light sources having lower intensity range while higher for the two other to create more shading effects. We have used cycle rendering with GPU support to render images in Blender [23]. The

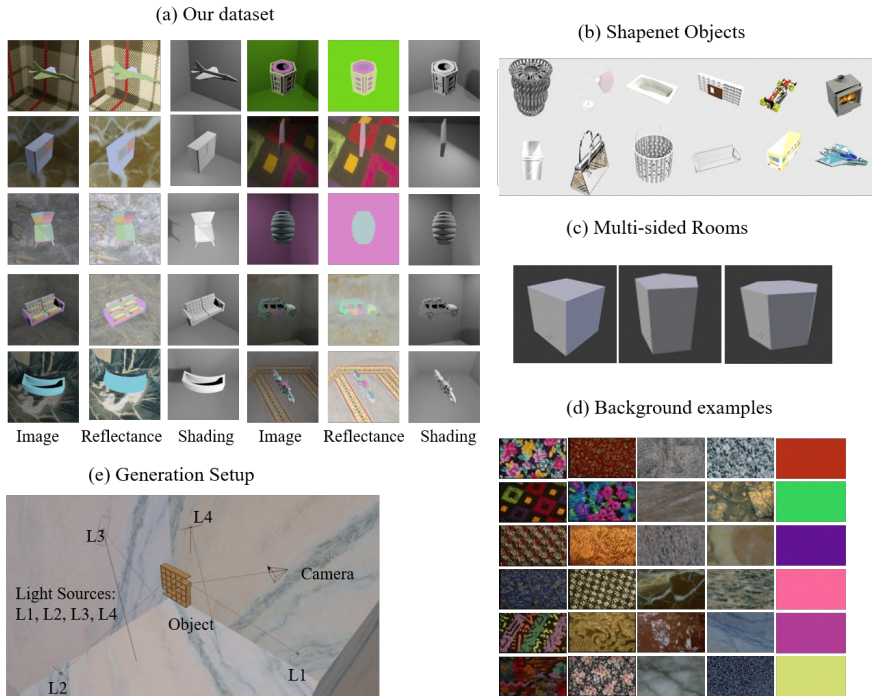


Figure 3.16: Dataset Generation Setup

dataset proposed in this chapter is a first version that will be extended in chapter 4 to estimate the light parameters and in 5 for a single image relighting by varying

the multiple parameters we have introduced above. The final scenes we create are somehow surreal, since images are presenting "*airplanes flying in closed rooms decorated with old-fashioned wall-papers*", but providing a wide range of realistic light effects with multiple color and shade interactions, thus we call it SID for *Surreal Intrinsic Dataset*.

The proposed dataset is specially designed for intrinsic decomposition. The way in which it is designed makes it to be easily tunable, since the number of parameters is not very high and the computation time for generation is very low. Therefore, new datasets can be derived for other applications like: (a) *Color constancy*, we just need to set the number of light sources we want and change the color, which requires to be stored in the ground-truth; (b) *Shadow removal*, we need to separately synthesize shading with and without shadows and make the network to learn the difference between this two images; and (c) *Specularity detection*, in this case we need to introduce non diffuse objects in the scene and generate their specular components for the ground-truth. Additionally, complexity could be increased by adding more than a single object, or inserting 3D textures on objects and walls, the scenes can be modeled depending on the light effect to be estimated.

### 3.3 Deep Neural Network

In this section, We propose a deep architecture based on the following three criteria: (a) the use of a U-NET-based, Encoder-Decoder, architecture, which has been the most usual [8, 110] and natural way to solve this pixel-wise regression problem; (b) increasing efficiency and speed properties by introducing split inception modules decreasing the number of parameters [116]; and (c) introducing physical constraints of the intrinsic decomposition model at the loss function like in [8, 69]. We will refer to our architecture as IUI as for *Inception U-Net* based for *Intrinsic* image estimation.

Network scheme can be seen in figure 3.17. It has one encoder and two decoder streams to predict shading and reflectance simultaneously. Encoder has five inception module followed by *conv + pool* layers to learn both shading and reflectance representations, and it has two decoders with 5 inception blocks followed by an upsampling layer. The output of the inception module in decoder is concatenated with respective encoder inception outputs. We used split-inception module in our network where the  $n \times n$  convolutional filters are substituted by  $n \times 1$  and  $1 \times n$  filters. This modification decreases overall learning parameters of the network that significantly results in a faster learning.

Inspired by the architectures presented in DARN [69] and IntrinsicNet [8] we define a physical loss that constraints the solution to fulfil the intrinsic image model. We propose a unique global loss as a weighted sum of three terms, given by

$$\mathcal{L}_{Int}(I, \hat{R}, \hat{S}) = \alpha_1 \mathcal{L}_{Ref}(R, \hat{R}) + \alpha_2 \mathcal{L}_{Sha}(S, \hat{S}) + \alpha_3 \mathcal{L}_{RS}(I, \hat{R} \cdot \hat{S}) \quad (3.8)$$

where the first two terms,  $\mathcal{L}_{Ref}$  and  $\mathcal{L}_{Sha}$ , are two terms ensuring that Reflectance,  $\hat{R}$ , and Shading,  $\hat{S}$ , predictions fit the ground-truth data, respectively. The third term,  $\mathcal{L}_{RS}$  forces the predictions to hold the intrinsic product model of Equation 1.9 and bound both outputs. All three terms are computing the mean square error (MSE) between the two inputs. It measures per pixel squared error between predicted images and the input ground-truth data,  $R$ ,  $S$  and  $I$ , for reflectance, shading and original image, respectively;  $\alpha_i$  are the corresponding weights for each term. Our proposed network can be easily extended by stacking more decoder streams for each output and introducing more constrains in the loss function.

We implemented our network on Keras [22]. Weights were initialized using He Normal [57]. We used Adam optimizer [60] with initial learning rate 0.0002 which is decreased with factor of 0.1 on reaching plateau. Since our network is all based on convolution layers, it enables to take any image size at the input. For our dataset the image size was decreased to  $256 \times 256$  and for Sintel[17] we used  $192 \times 448$ . We used batch size of 8 for all experiments and used three dropout layers per decoding stream with 50% dropout rate. Our intrinsic loss computes mean square error between the real and the predicted shading and reflectance, and the product of these two with the original image. We initialized with equal weights for all  $\alpha_i$  in our loss function. Network was trained from scratch with our dataset and fine tuned for others like MPI Sintel [17] and MIT [44]. We also tested our network on IIW [12] images and provided results just for qualitative purpose.

We randomly split our dataset in 60% of images used for training and 40% used for testing. We used 50 epochs for computing the loss in this experiment. This split size could require to be changed if we would change the composition of the dataset by using a different set of parameters in the image generation stage. Thus, in all experiments based on our dataset, we trained the IUI-Network on 15,000 images and tested on 10,000 images. We compute the performance of our architecture on MPI Sintel [17] and MIT [44]. Due to their small dimension or high redundancy, training a deep model directly on them is not feasible. To compare the feasibility of our approach with previous ones, we test on the existing datasets after specifically fine-tuning our trained IUI-Network.

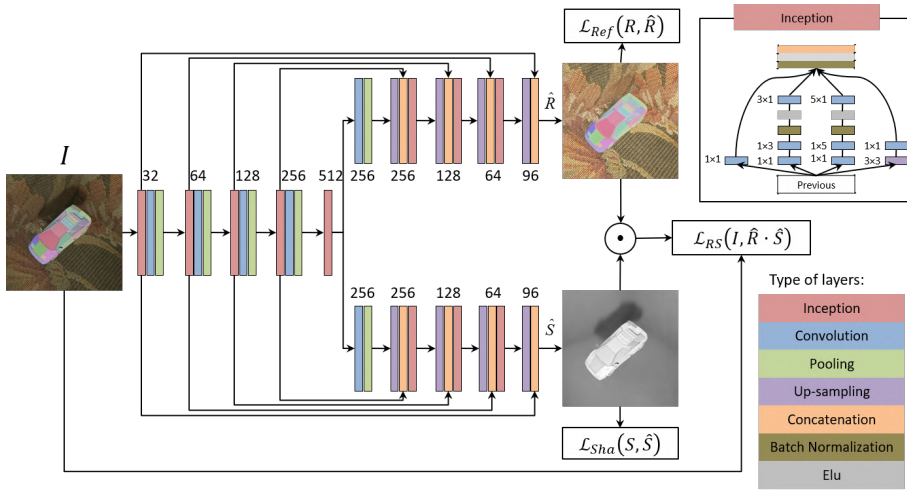


Figure 3.17: IUI-Network architecture. One encoder and two decoders for reflectance and shading estimation. Three inter-related loss functions. Type of layers are indicated by a color code given at right-bottom of the figure. Scheme of inception modules are given at left-top of the figure.

### 3.4 Experiments and Results

In this section, we evaluate our IUI-Network architecture trained on our proposed dataset. We show that the proposed framework formed by a flexible architecture trained on a synthetic dataset with a wide range of variations helps to increase the quality of the results on generic intrinsic image estimation. To evaluate the approach, we compare our performance on existing datasets, either synthetic or real, and we compare to different methods. We show that with this preliminary dataset we achieve state of the art results.

In the following sections we perform 5 different experiments. In experiment 1, we show the results of our framework trained and tested on our dataset overpasses the performance of Retinex algorithm [44], which is a baseline method. In experiment 2 we provide the results on MIT dataset, which is only formed by real images. Results show that our fine-tuned network is able to get near state of art results. Similarly, in experiments 3 we show the effectiveness of our IUI network as it gives good results on Sintel dataset. In experiment 4 we provide a qualitative comparison on IIW dataset, results show that our network is able to capture texture

details better than other methods. Finally, in experiment 5 we test the performance of the IUI architecture on Shapenet-Intrinsic dataset independently of our dataset

### 3.4.1 Experiment 1. Our dataset

In this first experiment, we compared IUI-Network with Retinex[44] for intrinsic decomposition. Results depicts that our network shows much improvement as compared to Retinex method. We computed the error separately on background walls, foreground objects and complete images. We show the results in table 3.1.

We can see that our IUI-Network considerably improves the predictions of intrinsic decomposition for all parts of images dividing the error by a factor that goes from 5 to 50. In general the error improvement is better for shading than for reflectance, since the dataset present a strong bias on flat walls presenting large areas of homogeneous shading. This agrees with the fact that the error decreases more on backgrounds than foregrounds. The main reason for a better performance on the background is that foreground object has more diversified texture and shape which results in more shading and reflectance variation rather than background.

As mentioned, we get the best improvements on backgrounds for all metrics except for DSSIM in shading. Our intuition for better DSSIM for foreground rather than background is because of shadows. They are a structural part of background shading and our network is better in getting the structural shading of the objects than of cast shadows. To improve the recovery of the cast shadow structure in the future, one possibility would be to diversify backgrounds using different textures for each wall and more diverse shapes than just flat walls. Introducing more randomly positioned multiple objects would reduce the bias that favour the background prediction and to use multiple color light source will make more challenging intrinsic image dataset for whole parts of image. We explored these options in the next versions (SID2 and SID3) of this dataset in 4 and 5.

In figure 3.18 we show qualitative results on our dataset, while our method is able to get correct intrinsic decomposition in most of images, it tends to have some errors in removing shadows from more plain background in reflectance images and keeping all shadow details in shading image, it also shows some blurring around room corners in shading images. First 3 images shows more succesful cases while last 3 shows some false intrinsic decomposition.

### 3.4. Experiments and Results

Method	Reflectance			Shading		
	MSE	LMSE	DSSIM	MSE	LMSE	DSSIM
Retinex (whole image)[44]	0.0500	0.049	0.17	0.0400	0.0403	0.24
IUI (foreground object)	0.0046 <i>(10.9)</i>	0.0038 <i>(12.9)</i>	0.029 <i>(5.9)</i>	0.0023 <i>(17.4)</i>	0.0020 <i>(20.2)</i>	<b>0.0178</b> <i>(13.5)</i>
IUI (background walls)	<b>0.0016</b> <i>(31.3)</i>	<b>0.0014</b> <i>(35.0)</i>	<b>0.019</b> <i>(8.9)</i>	<b>0.0010</b> <i>(40.0)</i>	<b>0.0008</b> <i>(50.4)</i>	0.023 <i>(10.4)</i>
IUI (whole image)	0.0020 <i>(25.0)</i>	0.0019 <i>(25.8)</i>	0.020 <i>(8.5)</i>	0.0011 <i>(36.4)</i>	0.0009 <i>(44.8)</i>	0.022 <i>(10.9)</i>

Table 3.1: Errors for reflectance and shading predictions on our dataset. Comparison between our IUI architecture and Retinex algorithm. IUI decreases the error of Retinex by the factor given in brackets. Errors are separately reported on object, on foreground and the on whole image.

#### 3.4.2 Experiment 2. MIT dataset

In this second experiment we evaluated IUI-Network on one of the most used dataset, MIT Intrinsic [44], and showed that it achieves nearly state of art results in all matrices. Similar to what all deep learning methods do, we fine-tuned our network from previous experiment to decompose the image in reflectance and shading, we used those weights as initialisation. Results on MIT dataset are given in table 3.2, the error is computed on the foreground mask to allow comparison with the rest of results reported in previous works.

Best performances are bold in the table. We achieve best errors for reflectance, and very close to IntrinsicNet [8] for Shading. These results confirm the generalization capability of our network and the effectiveness of training on our synthetic dataset that is able to generalize on realistic images. In this case, our network was able to get better DSSIM results that shows it is able to learn texture and edges information better than other methods.

In figure 3.19 we show some results for visual comparison with other state of art methods. Our estimation of reflectance is giving correct and sharp color information in dark shaded image areas. Regarding shading images we present a very good estimation for the turtle, and a worse one for the frog, where we can see some blurring effects and some shape details are lost.

Method	Reflectance			Shading		
	MSE	LMSE	DSSIM	MSE	LMSE	DSSIM
Retinex[44]	<b>0.0032</b>	0.0353	0.1825	0.0348	0.1027	0.3987
SIRFS[5]	0.0147	0.0416	0.1238	0.0083	0.0168	0.0985
Direct Intrinsic[87]	0.0277	0.0585	0.1526	0.0154	0.0295	0.1328
ShapeNet[110]	0.0278	0.0503	0.1465	0.0126	0.0240	0.1200
CGIntrinsic[71]	0.167	0.0319	0.1287	0.0127	0.0211	0.1376
IntrinsicNet[8]	<b>0.0051</b>	0.0295	0.0926	<b>0.0029</b>	<b>0.0157</b>	<b>0.0441</b>
RetiNet[8]	0.0128	0.0652	0.0909	0.0107	0.0746	0.1054
IUI	<b>0.0046</b>	<b>0.0197</b>	<b>0.054</b>	0.0038	0.020	0.0557

Table 3.2: Estimation errors on MIT dataset reported in previous works by different methods and for our IUI architecture.

### 3.4.3 Experiment 3. Sintel dataset

Similarly to previous experiments, here we evaluate the performance of our architecture on the Sintel dataset [17]. We followed the approach introduced by Narihira *et al.* in [87] and provide the results for both image-split and scene-split tests. In image-split the whole 890 images of the dataset are randomly assigned, such that 50% are used for training and the rest for testing. In scene split, training and testing data are decided on the basis of scenes, rather than images, which makes it more difficult in terms of capability to generalize.

In table 3.3 we can see the results of our network fine tuned on the image split test, denoted as *IUI fine-tuned on CS*, altogether with previously reported results. In this case we can see that our method is giving an error very close to the state of the art method at that time which was Fan *et al.* [32], we highlight those numbers closer by less than 0.002 to the best reported error. In the same table we report the results of two more experiments. We denote as *IUI without fine-tuning* the results of testing our network without any training on this test set, just to see the performance of our network just trained on our SID, in this case we can see that our approach is better than most of previous methods before Fan *et al.* method. Finally, we give the results of one additional experiment, we refer to it as *IUI fine-tuned on GLS*, in this case we test our network fine tuned on a new ground-truth. Considering that Sintel dataset is based on synthetic images where shading is colored, we built a new version of this dataset with a grey-level shading that fully accomplishes the dot product model of intrinsic image decomposition with a single channel shading and RGB reflectance. We do this test to evaluate how our network is performing on this constraint, since it is trained under a loss function that forces the product

model to be fulfilled ( $\mathcal{L}_{RS}$  loss) like in SID dataset. We can show that in this case, our network is overcoming the state of the art in reflectance estimation.

In table 3.4 we can see the results of our network tested in the same manner as before but now on the scene split test. We confirm similar results. We are close to state of the art when fine-tuned on the this Sintel test and we overcome them in reflectance when we fine-tune on a grey-level shading dataset.

In figures 3.20 and 3.21 we show some examples of our results for image split and scene split respectively. We can see that Fan *et al.* and our IUI are showing the best results from a qualitative point of view, getting a sharp and correct intrinsic decomposition. We can observe some interesting details in these images. For example, estimated reflectance in image split figure 3.20 our network is giving better sky, clouds and background wall reflectance decomposition as compared to all other methods, and we get correct rust effects on cart and right brownish hairs for women. Regarding, scene split experiments, our network is getting better details in some parts of the image, for example in figure 3.21 estimated reflectance in left column is correctly separating most shading effects on background walls, floor and cast shadows. We also get correct reflectance for animal left and right eyes. Similarly, in right column, our estimation is getting best recovery of color details in teeth, mouth and arm colors.

However, our method does not perform accurately in recovering shading images and it shows overall false color for shading detection. We believe the main reason behind this issue is that our method is trained on our current SID version that only contains white light sources, while Sintel scenes present some unnatural lights with bias towards bluish or brownish colors. This opened a further research line in our framework which is introducing more color light sources in SID. Finally, we also want to point out, that on the shading of left column image in figure 3.21, our method recovers some specular effects on animal face which, less artifacts on floor area and a smoother background, all closer to the ground-truth shading.

#### 3.4.4 Experiment 4. IIW dataset

In this section we provide a qualitative experiment on real images. Although, this is a very preliminary experiment, since we did not focus on training our approach on this dataset, we want to add here two interesting observations: (a) the main strength of our architecture is the ability of recovering texture information in the reflectance component; and (b) the proposed architecture is based on a very simple and versatile approach that presents a very low computational cost compared to



### Chapter 3. Reflectance and Shading Estimation

Method	Reflectance			Shading		
	MSE	LMSE	DSSIM	MSE	LMSE	DSSIM
Retinex[44]	0.0606	0.0366	0.227	0.0727	0.0419	0.24
Lee <i>et al.</i> [68]	0.0463	0.02224	0.199	0.0507	0.0192	0.177
SIRFS[5]	0.042	0.0298	0.21	0.0436	0.0264	0.206
Chen and Koltun[20]	0.0307	0.0185	0.196	0.0277	0.019	0.165
Direct Intrinsic[87]	0.01	0.0083	<b>0.02014</b>	0.0092	0.0085	0.1505
Fan <i>et al.</i> [32]	<b>0.0069</b>	<b>0.0044</b>	<b>0.1194</b>	<b>0.0059</b>	<b>0.0043</b>	<b>0.0822</b>
IUI fine-tuned on CS	<b>0.0072</b>	<b>0.0054</b>	0.1374	0.0068	<b>0.0059</b>	0.1247
IUI without fine-tuning	0.023	0.015	0.21	0.035	0.022	0.255
IUI fine-tuned on GLS	<b>0.0062</b>	<b>0.0047</b>	0.1297	<b>0.0057</b>	<b>0.0048</b>	0.1183

Table 3.3: Results on Sintel Image Split dataset. Best errors are highlighted in bold.

Method	Reflectance			Shading		
	MSE	LMSE	DSSIM	MSE	LMSE	DSSIM
Direct Intrinsic[87]	0.0238	0.0155	0.226	0.0205	0.0172	0.1816
Fan <i>et al.</i> [32]	<b>0.0189</b>	<b>0.0122</b>	<b>0.1645</b>	<b>0.0171</b>	<b>0.0117</b>	<b>0.1450</b>
IUI fine tuned on CS	0.0213	0.0140	0.1787	0.0253	0.01721	0.1874
IUI without fine-tuning	0.023	0.0154	0.20	0.034	0.023	0.24
IUI fine tuned on GLS	<b>0.01733</b>	<b>0.0110</b>	<b>0.16189</b>	0.0201	0.013182	0.1618

Table 3.4: Result on Sintel Scene Split dataset. Best errors are highlighted in bold.

the rest of state of art methods.

In figure 3.22 we show an example to compare between one of most recent and efficient methods with real images, which is Fan *et al.* [32] trained on this dataset; and our IUI architecture without fine tuning on the dataset. The predicted images shown in the figure are by direct application of IUI trained on our SID dataset. In this example we can see that at the level of the object textures, our approach is providing a better decomposition, reflectance prediction is capturing all the texture details of blanket, carpet and curtain, while we fail in getting unsharp edges for these objects. However, the decomposition provided by Fan *et al.* results is getting very good sharp versions of Shading while including a lot of details that should be in the reflectance image. Therefore, some more work is required in setting more adapted versions of our dataset to overcome these problems, but, we think our framework is providing an excellent tool to overcome the difficulties of a good intrinsic decomposition at all levels.

In table 3.5 we show two quantitative aspects we can conclude from experiments on IIW dataset. We have to remind that this dataset is based on human judgments about shading and it is providing poor information about true reflectance and its

spatial coherence. Although our network was not trained for IIW dataset, and we have proved our approach is better in getting reflectance estimation, still we get 22.50% accuracy between our shading predictions and those provided by human judgments. In the same table, we also list computation times for several methods and we show that our inception-based network, and thanks to its small number of parameters, is giving the second best performance. This table numbers have been reproduced from [32]

Method	WHDR(mean)	runtime(sec)
Shen <i>et al.</i> [108]	36.90	297
Retinex(color)[44]	26.89	198.5
Retinex(gray)[44]	26.84	225.3
Graces <i>et al.</i> [38]	25.46	5.1
Zhao <i>et al.</i> [134]	23.20	34.7
IUI ( <i>without fine-tuning</i> )	22.50	<b>0.02</b>
L1 flattening[14]	20.94	310.94
Bell <i>et al.</i> [12]	20.64	214
Zhou <i>et al.</i> [136]	19.95	300
Nestmeyer <i>et al.</i> (CNN)[89]	19.49	<b>0.006</b>
Nestmeyer <i>et al.</i> [89]	17.69	300.086
Bi <i>et al.</i> [14]	17.67	300
Fan <i>et al.</i> [32]	<b>14.45</b>	0.1

Table 3.5: Result on IIW dataset

### 3.4.5 Experiment 5. Evaluating IUI architecture

The goal of this experiment is to test the performance of the IUI architecture independently of our dataset. We trained our IUI-Network from scratch on the ShapeNet-Intrinsic dataset introduced by Baslamisli *et al.*. We followed the same training/testing split introduced in their work and we used the same metrics to compare performance. Results reported in table 3.6 show that our architecture and the dataset help to improve results. In the second experiment, we trained and tested our IUI-Network on the ShapeNet-Intrinsic dataset from scratch. Results show that our network outperforms state of art methods on this specific dataset. Best performances are bold faced in the table.

Additionally, we performed another experiment that we called *IUI without fine-tuning*. We trained our network on our own SID dataset and tested it on

ShapeNet-Intrinsic dataset. Our network without fine-tuning performed better than Direct-Intrinsics[87] fine-tuned on this dataset. This experiment, likewise those reported in previous sections, shows a good generalization capability of our IUI network combined with our SID dataset.

Method	Reflectance			Shading		
	MSE	LMSE	DSSIM	MSE	LMSE	DSSIM
Direct Intrinsic [87]	0.1487	0.6868	0.0475	0.0505	0.3386	0.0361
ShapeNet [110]	0.0023	0.0349	0.0186	0.0037	0.0608	0.0171
IntrinsicNet [8]	0.0005	0.0072	0.0909	0.0007	0.0505	0.0084
RetiNet [8]	0.0003	0.0205	0.0052	0.0004	0.0253	0.0064
IUI <i>trained on Shapenet-Intrinsic</i>	<b>0.0002</b>	<b>0.0193</b>	<b>0.0032</b>	<b>0.0003</b>	<b>0.0229</b>	<b>0.0047</b>
IUI <i>without fine-tuning</i>	0.0073	0.1926	0.0396	0.0479	0.2324	0.0291

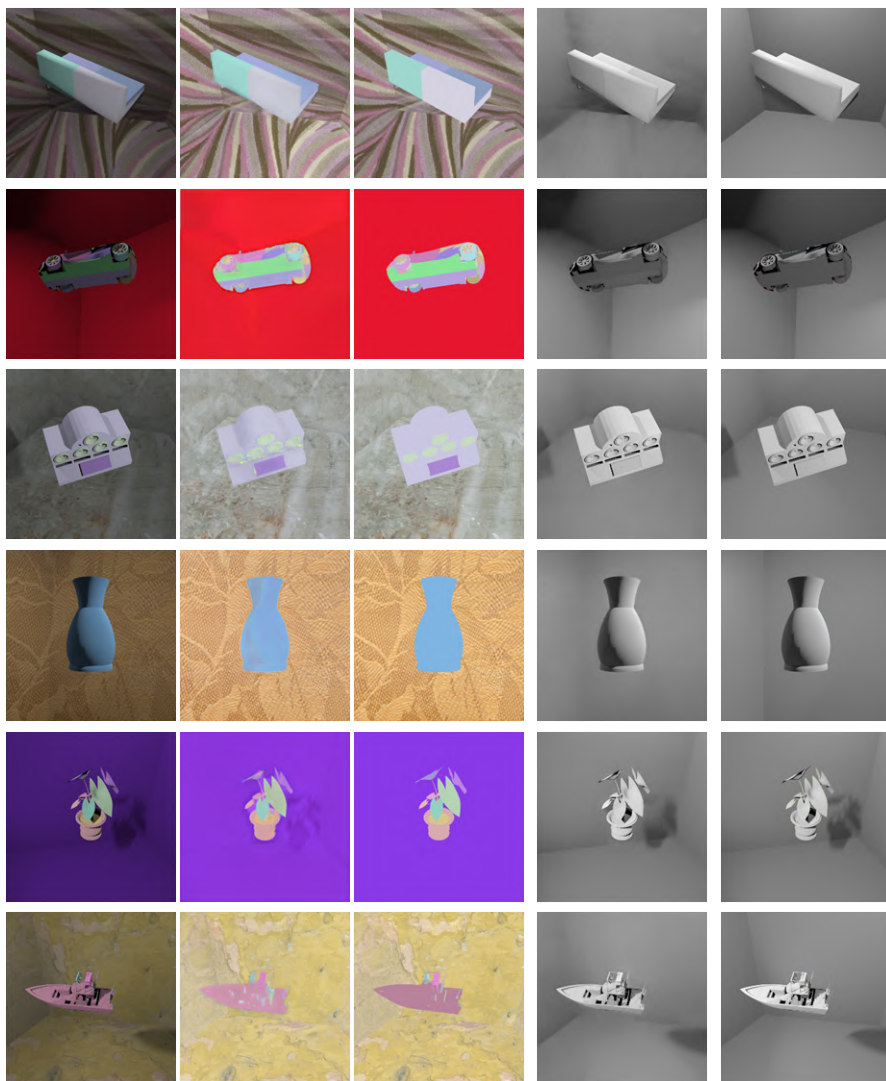
Table 3.6: Estimation errors of different architectures trained on Shapenet-Intrinsic dataset. In bottom row the errors of IUI architecture trained on our dataset and tested on Shapenet-Intrinsic.

## 3.5 Conclusion

The importance of accurate and large datasets in this data-driven deep-learning era is undeniable. In this chapter, we focused on creating such a large intrinsic image dataset. We described different possible ways to create ground-truth for this task with their pros and cons. First, we proposed a color-based data augmentation technique that extends the training data by increasing the variability of chromaticity and preserving the reflectance geometry of the ground-truth. In this way, the lack of data can be partially solved with data augmentation. Secondly, We presented a pipeline to build a dataset by registering synthetic with acquired scenes to be able to automatize the process of building the dataset ground-truth. This is a really hard task that requires geometric and photometric calibration between two worlds. We used this hybrid ground-truth creation technique to create a Doc3Dshade dataset in chapter 5 to remove illumination effects from document images. Lastly, we introduced a completely synthetic dataset of 25,000 images named SID for *Surreal Intrinsic Dataset*[112]. This dataset has a lot of variations of shading and reflectance effects.

In parallel with the creation of datasets, we propose a versatile framework to define and train a convolutional network able to perform an intrinsic decomposition through training on a dataset with a large variety of light effects and color reflectances. our proposed CNN architecture has been defined in a simplistic way

to reduce its number of parameters and enough flexible to be adapted to multiple types of visual tasks related to light effect estimation. The results obtained by all the experiments we report in this chapter, make us to be optimistic about the capabilities of the presented approach to train networks devoted to solve task related to the estimation of light effects. In all the reported experiments we show a performance close to the state of the art of the problem of intrinsic decomposition in shading and reflectance.



(a) Input image (b) Ref. Predict. (c) Ref. GT (d) Shad. Predict. (e) Shad. GT

Figure 3.18: Some examples of our SID dataset. (a) Original Images. (b) and (d) are Reflectance and Shading estimation by our IUI network, respectively. (c) and (e) are GT Reflectance and Shading, respectively.

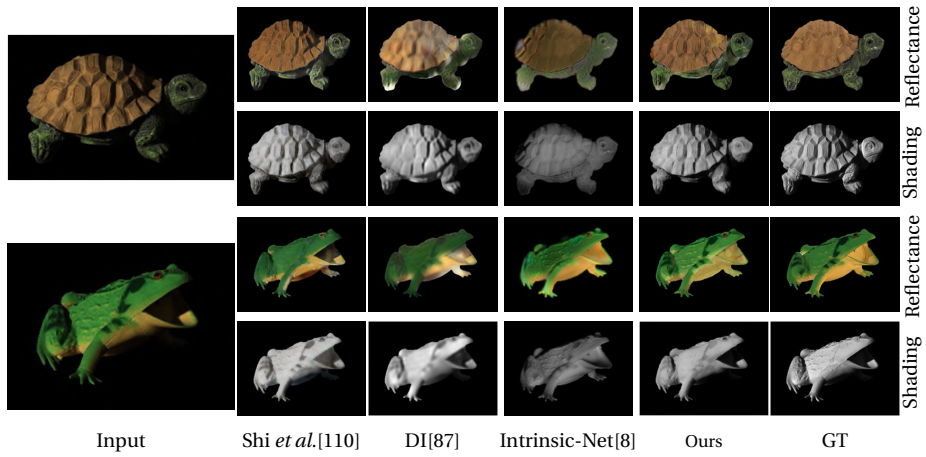


Figure 3.19: Qualitative Results on MIT intrinsic image dataset, compared to other methods, we achieved sharp and better colors and removed shading effects. Our method performed best in bringing reflectance details from dark part of image.



Figure 3.20: Visual comparison on MPI-Sintel dataset using image split.



Figure 3.21: Visual comparison on MPI-Sintel using Scene split.

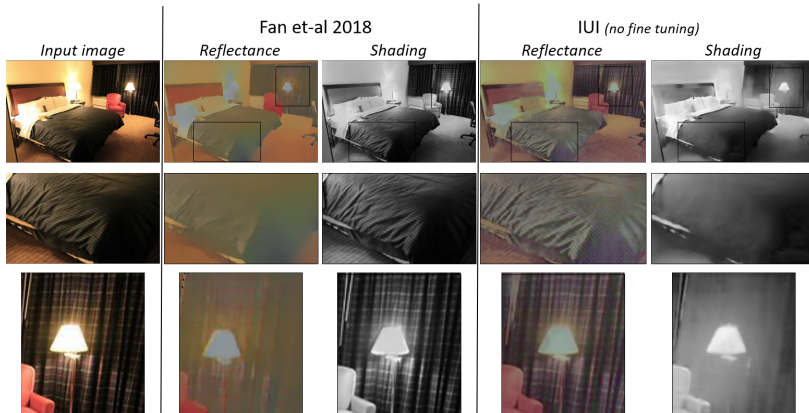


Figure 3.22: Qualitative results on IIW





## 4 Light Source Estimation

In this chapter, we focus on another intrinsic image modality to estimate light source properties from a single image. As an initial step in this domain, we present a method to estimate the direction and color of a scene light source from a single image. Our method is based on two main ideas: (a) we use a new synthetic dataset with strong shadow effects with similar constraints to *SID dataset*; and (b) we define a deep architecture trained on the mentioned dataset to estimate direction and color of the scene light source. Apart from showing a good performance on synthetic images, we additionally propose a preliminary procedure to obtain light positions of the Multi-Illumination dataset, and in this way, we also prove that our trained model achieves a good performance when it is applied to real scenes.

### 4.1 Introduction

Scene appearance is directly dependent on the light source properties, such as the spectral composition of emitted light, and the position and direction of the light source, whose interaction with the scene objects provoke shaded surfaces or dark cast shadows that become essential visual cues to understand image content, these effects are discussed in more detail in chapter 1. Estimating the properties of the light conditions from a single image is an initial step to improve subsequent computer vision algorithms for image understanding. In this task, we perform a preliminary study to estimate color and position of light in a simple and unified approach, that is based on the shading properties of the image where we assume a single scene illuminant.

Estimating the color of the light from a single image has been focus of attention in previous research. Computational color constancy (CC) has been studied in a large number of works [33, 34, 42] where the problem was tackled from different points of views [42]. A first approach was to extract statistics from RGB image values under different assumptions to estimate the canonical white. A second approach was to introduce spatio-statistical information like gradients or frequency content of the image. One last group of CC algorithms was to try to get the information from physical cues of the image (highlights, shadows, inter-reflections, etc). In the last years, new approaches have been based on deep learning frameworks where the solution is driven by the data with physical constrains in the loss functions. An updated comprehensive compendium and comparison of CC algorithm performances can be found in [21, 25, 77, 130]. Our proposal is also based on a deep architecture, but color of the light source is jointly estimated with the light direction.

Estimating the direction of the light has also been tackled from different areas like, computer graphics, computer vision or augmented reality. Single image light direction estimation can be divided in two different kinds of approaches. First, those in which light probes with known reflectance and geometric properties are used. A specular sphere is commonly used to represent light position in different computer graphics applications [1, 28, 101]. But, random shaped objects to detect light position were used by Mandl et-al in [80], jointly with a deep learning approach to get the light position with each of these random shaped object. Second, we find those works in which no probe is used, and where multiple image cues such as shading, shape and cast shadows are the basis to estimate light direction. Some examples of these works can be found in computer vision literature [2, 55, 92, 93, 100].

Some deep learning methods have been proposed to estimate scene illumination and have been used for different computer graphics tasks. Gardner et al.[40] introduced a method to convert low dynamic range (LDR) indoor images to high dynamic range (HDR) images, first they used a deep network to localize the light source in LDR image environmental map and then they used another network with these annotated LDR images to convert them to HDR image. Following a similar approach, Geoffroy et al. [49] introduced a method to convert outdoor LDR images to HDR images. They trained their network with a set of panorama images and predicted HDR environmental maps with sky and sun positions. Later on, Geoffroy et al [39] extended their previous idea for indoor lighting but replaced the environmental maps with light geometric and photometric properties. Sun et al. [115] introduced an encoder-decoder based network to relight a single portrait image, the encoder predicts the input image environmental map and an embedding for

the scene information, while the decoder builds a new version of the image scene with the target environmental map and obtains a relighted portrait. More recently, Muramann et al. [86] have introduced the *Multi-illuminant dataset* of 1000 scenes each one acquired under 25 different light position conditions, and they used a deep network to predict a right sided illuminated image from its corresponding left sided illuminated image. We also evaluated our proposal for light source properties estimation on this new wild dataset after providing a procedure to compute the light direction from each sample.

To sum up, we can state that a large range of works have tackled the problem of estimating color and direction of the scene light source from different points of views and focusing on specific applications. In this activity, we propose an easy end-to-end approach to jointly characterize the light source of a scene, both for color and direction. We pursuit to measure the level of accuracy we can achieve, in order it can be applied to a wide range of images, without using probes in the scene and becoming a robust preliminary stage to be subsequently combined with any task.

Rest of the chapter is organized as follows: first we introduce a new synthetic dataset in section 4.2, secondly we use it to train a deep architecture that estimates light properties from a single image in section 4.3, finally we show how our proposal performs on three different scenarios, synthetic, real indoor and natural images in section 4.4.

## 4.2 SID2 Dataset

In order to train our end-to-end network that estimates light conditions we developed a newer version of SID1 dataset (presented in section 3.2.4), which was created for intrinsic image decomposition. This dataset is formed by a large set of images of surreal scenes where Shapenet objects [19] are enclosed in the center of multi-sided rooms with highly diverse backgrounds provided by flat textures on the room walls, resulting in a large range of different light effects. We proposed a new version of SID1 dataset better adapted to this problem, using the same methodology and software used in SID1, which is based on an open source Blender rendering engine to synthesize images.

Our new dataset is called SID2 dataset. The main difference with its preceding version is that it introduces more than one object in each scene, with the aim of increasing the number of strong light effects and interactions. Additionally, we

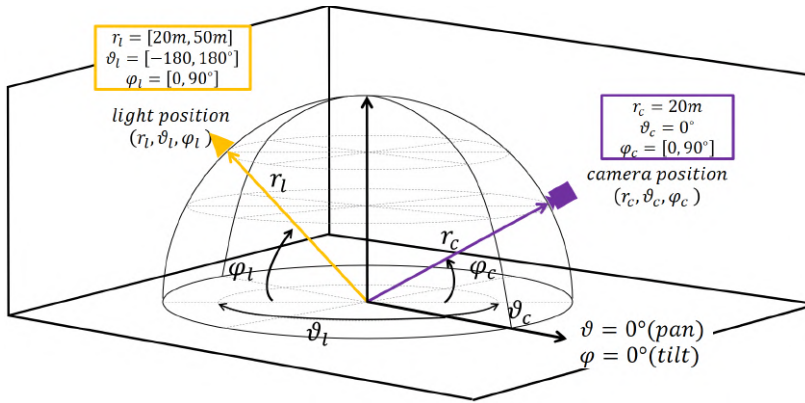


Figure 4.1: Image Generation Setup. Camera and light positions are given in spherical coordinates  $(r, \vartheta, \varphi)$ .

also introduce more variability in the distance from the light source to the scene center. The dataset is formed by 45,000 synthetic images with the corresponding ground-truth data: direction and color of the light source.

We did several assumptions in building the dataset: (a) objects are randomly positioned around the scene center but always close to the room ground floor to have realistic cast shadows; (b) light source direction goes from scene center to a point onto an imaginary semi-sphere of a random radius and with random RGB color; (c) camera is randomly positioned at a random distance from the center of the scene and always with the focal axis pointing to the scene center. In figure 4.1 we show the diagram of the synthetic world we defined for the generation of SID2. More specifically, we took 45,000 3D objects from Shapenet dataset [19]. In this dataset we also used diffuse bidirectional scattering distribution function (BSDF) with random color and roughness values for each mesh texture in each object. This roughness parameter controls how much light is reflected back from each object surface. We randomly picked from 1 to 3 objects in each image. They were placed at random locations within the camera view range. We placed an empty object in the center of the scene to ensure non-overlapping between the rest of objects. Light direction was randomly defined in spherical coordinates  $(r_l, \vartheta_l, \varphi_l)$ , being radius, pan and tilt, respectively. We took random values within the ranges of  $[20m, 50m]$   $[30^\circ, 90^\circ]$  and  $[0^\circ, 360^\circ]$  respectively, in steps of  $1^\circ$  for pan and  $5^\circ$  for tilt. Light intensity and chromaticity was randomly selected, but chromaticity was set using

color Temperature<sup>1</sup> to simulate natural lighting conditions. Camera position is also denoted in spherical coordinates as  $(r_c, \vartheta_c, \varphi_c)$ , where  $r_c$  was fixed at  $20m$  and pan,  $\vartheta_c = 0^\circ$ , the tilt range randomly varied within  $[10^\circ, 70^\circ]$ . In the final ground-truth (GT), light pan and tilt are provided with reference to the camera position, in order not to depend on real world positions which are usually not available in real images. Backgrounds were generated in the same way as in SID1.

### 4.3 Network Architecture

We propose an inception-based encoder-decoder architecture to predict light parameters. In figure 4.2 we give an scheme, where we can see that our encoder has five modules combining 3 types of layers: inception, convolution and pooling. The encoder input is the image that is transformed to a higher dimensional feature space, from which three decoders convert this embedding to a common feature space of pan, tilt and color of light source. Pan and tilt output predictions are given as functions of angle differences. We use the functions  $\sin(\vartheta_c - \vartheta_l)$  and  $\cos(\vartheta_c - \vartheta_l)$  to bound the pan output. Similarly, tilt prediction is represented as difference of angles  $\sin(\varphi_c - \varphi_l)$  and  $\cos(\varphi_c - \varphi_l)$ . Finally color is predicted here as R, G and B values. We used the split inception module from [116], which replaces  $n \times n$  convolution filters with  $1 \times n$  and  $n \times 1$  filter, to achieve faster convergence with overall less parameter. Our global loss function to estimate illumination parameters is based on three terms:

$$Loss(x, \hat{y}) = \alpha_1 \mathcal{L}_{pan}(x, \hat{y}) + \alpha_2 \mathcal{L}_{Tilt}(x, \hat{y}) + \alpha_3 \mathcal{L}_{Color}(x, \hat{y}) \quad (4.1)$$

where  $x$  is the input image,  $\hat{y}$  is a 7 dimensional vector giving the estimation of the scene light properties represented by  $x$ ,  $\alpha_i$  are the weights for the different loss terms defined for pan, tilt and color, and which are respectively given by:

$$\begin{aligned} \mathcal{L}_{Pan}(x, \hat{y}) &= MSE((\hat{y}_1 - \sin(\vartheta_c^x - \vartheta_l^x)) + (\hat{y}_2 - \cos(\vartheta_c^x - \vartheta_l^x))) \\ \mathcal{L}_{Tilt}(x, \hat{y}) &= MSE\{(\hat{y}_3 - \sin(\varphi_c^x - \varphi_l^x)) + (\hat{y}_4 - \cos(\varphi_c^x - \varphi_l^x))\} \\ \mathcal{L}_{Color}(x, \hat{y}) &= \arccos((\hat{y}_5 \cdot x_{RGB}) / \|\hat{y}_5\| * \|x_{RGB}\|) \end{aligned} \quad (4.2)$$

$\mathcal{L}_{Pan}$  and  $\mathcal{L}_{Tilt}$  are computed as the mean square error (MSE) between the estimations for pan,  $\hat{y}_1$  and  $\hat{y}_2$ , and for tilt,  $\hat{y}_3$  and  $\hat{y}_4$ , and a function of the difference between the camera and light positions for the ground-truth of  $x$ . The third loss term,  $\mathcal{L}_{Color}$ , is the mean angular error between the estimated RGB values,  $\hat{y}_5$ , and

---

<sup>1</sup>Temperature corresponds to the  $T$  parameter of the Planckian formula, used to describe the spectral wavelength decomposition of black-body radiators light sources types.

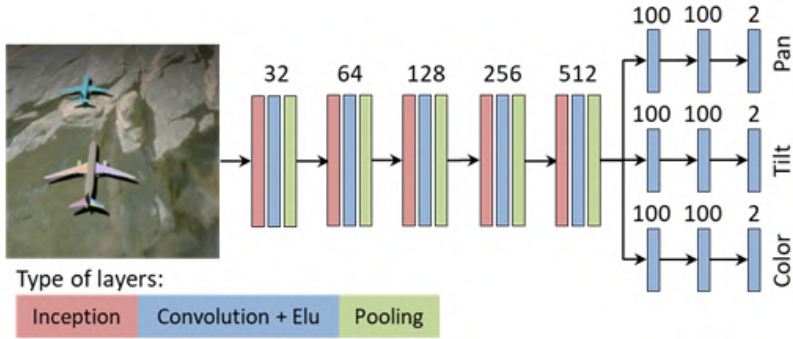


Figure 4.2: Deep Architecture. Inception module from [116]

the color of the light for  $x$  image provided in the ground-truth.

Our network has been trained using Adam optimizer [60], with initial learning rate 0.0002 which is decreased with factor of 0.1 on reaching plateau. Weights are initialized using He Normal[57]. All experiments in next sections were trained using a batch size of 16. In the following sections we show the results of several experiments to evaluate the architecture performance on different datasets and conditions.

## 4.4 Experiment and results

### 4.4.1 Synthetic Dataset

In this first experiment we trained and tested the proposed architecture on two different datasets: SID1 (single object) and SID2 (multiple objects). The results are shown in table 4.1, where we separately compute different angular errors. Direction error is given separately in the pan and tilt components, and the global angular error for direction estimation. We can see that all the estimations are improved when the network is trained on a more complex dataset, like SID2, where multiple objects light interactions provide richer shading cues. However, there is slight improvement for color estimation, performance is similar for both datasets. In figure 4.3 we show qualitative results on SID2 dataset. Images are ordered from smaller (left) to larger (right) direction estimation error.

Dataset	Pan	Tilt	Direction	Color
SID1	14.63	9.86	16.98	1.05
SID2	10.46	9.21	14.22	1.02

Table 4.1: Estimation Errors (in degrees) for light source direction and color with the proposed architecture trained on SID1 and SID2.

Intuitively as the light becomes more zenithal, the shadows shorten and cast shadows present more uncertainty to estimate light direction. We analyzed the performance of the method at different tilt locations of the light source in the input image, from the ground (level 1:  $[30^\circ, 50^\circ]$ ) up to the zenithal area (level 3:  $[70^\circ, 90^\circ]$ ). This effect is confirmed in table 4.2, where estimated errors in direction clearly increase from level 1 to level 3.

Tilt range	Pan	Tilt	Direction	Color
Level 1	4.92	5.14	7.57	1.04
Level 2	8.51	5.14	11.97	0.90
Level 3	22.36	18.33	28.33	1.00

Table 4.2: Estimation Errors (in degrees) for light direction and color at different tilt levels.

Similarly, we analyzed the performance at different pan levels, each level covers  $90^\circ$  of pan area. Level 1 is when light comes from center front, level 2 from right, level 3 from back and level 4 is when light comes from left side. Tilt angle was kept between  $30^\circ$  and  $70^\circ$  to analyze pan error while minimizing zenithal tilt error effects. table 4.3 shows results for this experiment, both direction and color error are consistent in all levels of pan.

#### 4.4.2 Multi-illumination Dataset

Once we have evaluated our method in synthetic images, we want to analyze whether it generalises for real images. We have tested our method on the *Multi-illuminant dataset* (MID) [86], that contains 1000 different indoor scenes, all of them containing a diffuse and a specular sphere at random locations. Light source is mounted above the camera and can be rotated at different predefined pan and



## Chapter 4. Light Source Estimation

---

Pan range	Pan	Tilt	Direction	Color
Level1	6.87	5.10	9.10	0.98
Level2	6.24	5.12	8.80	0.96
Level3	6.35	5.09	9.46	1.03
Level4	6.01	5.16	9.03	0.97

Table 4.3: Estimation Errors (in degrees) for light direction and color at different pan levels.

tilt angles, creating different light conditions. The dataset provides the orientation of the light source for each acquired image, but since the light can bounce off the walls, the direction of the incident light on the scene is not defined by the light source angles and it needs to be recomputed.

We have defined a procedure to compute the incident light direction from the specular sphere present in all the scenes, whose highlights provide enough information to collect our GT data (tilt and pan angle between light and camera). The color of the light is obtained from the average color of the diffuse sphere. To obtain the light direction we used the similar ideas presented in 3.2.3 based on the fact that the angle of incident light is equal to the angle of outgoing light at the specular highlight on a spherical ball. We use a reference image in each scene where light and camera both are pointed in the same direction towards the center of the scene. We also assumed that the light is mounted at  $10^\circ$  height with respect to scene center. The angles obtained from this reference images allow to correct the angle displacement due to the sphere position shifting inside the image on the rest of scene images.

Dataset (Error)	Pan	Tilt	Direction	Color
Masked MID (Mean)	21.38	10.14	22.72	0.63
Masked MID (Median)	13.74	7.64	17.80	0.40
UnMasked MID (Mean)	14.28	6.96	15.44	0.36
UnMasked MID (Median)	8.20	4.83	10.83	0.24

Table 4.4: Estimation errors in degrees on two versions of MID dataset (with Masked or UnMasked spheres).

Starting from the network trained on SID2 it was fine tuned on this dataset

under two different conditions: a) keeping the reference spheres in the image, and b) masking them. Although specular spheres are not present in real images, the first configuration should provide an upper bound of our method performance on wild images. Table 4.4 shows the results on this experiment. As expected, network performance is much higher when complete images are used as inputs. We can also observe that results on color estimation are better than on SID2, mainly due to the stability of single white light source in the dataset. To analyze the results removing the influence of the outliers, we also reported median error on this dataset. Results are as good as the ones obtained only using synthetic images on the upper bound. Qualitative results are provided in figure 4.4. Top row depicts the original image, second row are the spheres generated from the GT information, and the third row shows the synthetic spheres generated with the obtained prediction.

Finally, we perform a last experiment on this dataset by dividing the test set in two: (a) images with incident light from the front, and (b) from the back. Table 4.5 also shows the errors computed for these two sets. Both color and direction errors are higher when the light comes from the back of the scene and a big area of the image becomes saturated. We want to note here that the GT we created present a low accuracy for the subset of images with back light sources. This is due to the inherent uncertainty derived from what can be inferred from spheres illuminated from the back. Therefore, this MID dataset division is highly recommended to analyze results derived from this GT.

Light position	Pan	Tilt	Direction	Color
Front	11.51	6.33	13.09	0.34
Back	32.90	11.21	31.23	0.52

Table 4.5: Estimation errors in degrees dividing MID dataset in front and back light.

### 4.4.3 Natural Images

Previous experiments show the performance of our method on synthetic and real indoor images. Here, we show a few qualitative results on real outdoor images. In figure 4.5 we show some examples with strong outdoor cast shadows, in order to visually evaluate the prediction we depict a synthetic pole at the left top corner. In these examples camera is assumed to be at  $45^\circ$  tilt from ground. Left side four images are from SBU shadow dataset [120] and the two on the right have been captured with a mobile device.

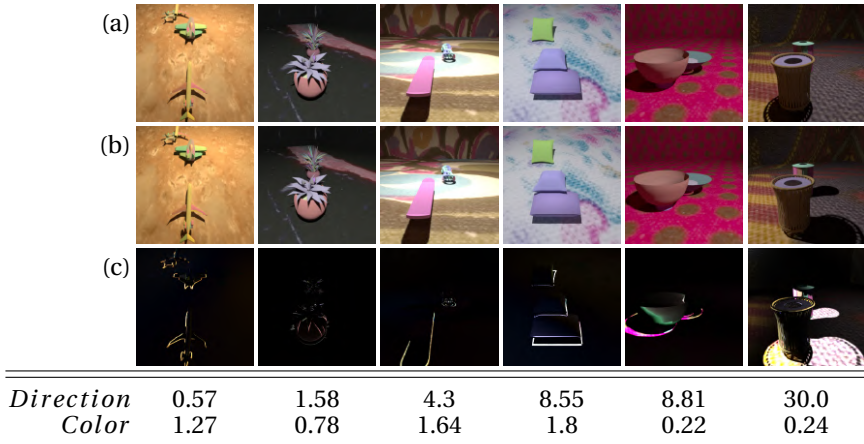


Figure 4.3: Direction and Color estimation examples on SID2 dataset: (a) Original images, (b) Generated images with estimated light properties, (c) RGB Image subtraction between (a) and (b). Bottom rows are the corresponding computed errors for direction and color in degrees, ordered from smaller (left) to larger (right) direction estimation error.

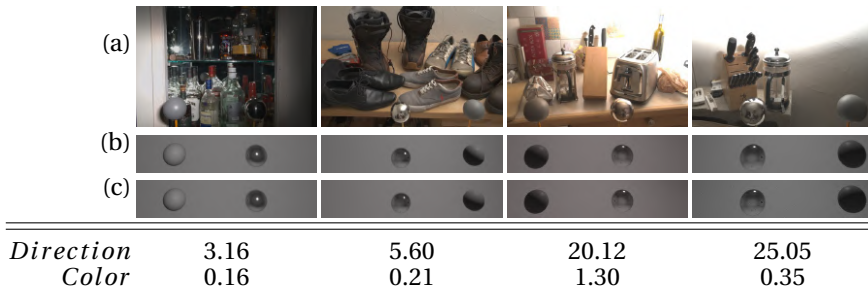


Figure 4.4: Direction and Color estimation examples on Multi-illumination dataset: (a) Original images, (b) Ground-truth plotted on corresponding spheres, (c) Estimations provided by our proposed architecture. Bottom rows are computed errors for direction and color in degrees.

## 4.5 Conclusions

In this chapter, we proved the plausibility of using a simple deep architecture to estimate physical light properties of a scene from a single image. The proposed

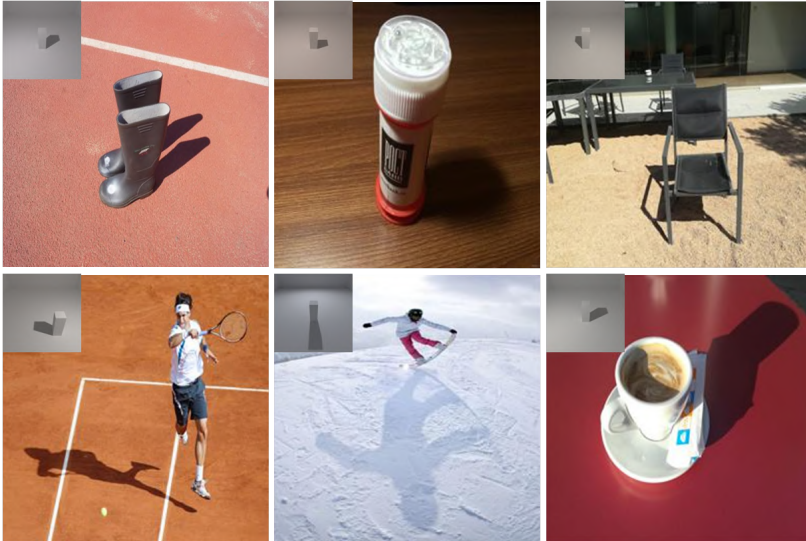


Figure 4.5: Examples of light direction estimation on natural images. Predicted direction is plotted top left in each image.

approach is based on training a deep regression architecture on a large synthetic and diversified dataset (SID2). We show that the obtained regressor can generalize to real images and we used this as a preliminary step for further complex tasks of a single image relighting in chapter 5.



# 5 Applications of Intrinsic Decomposition

In this chapter we propose to use the previous results and experience in decomposing intrinsic image properties for two different applications: (a) Removing light effect in document images; and (b) Relighting a scene from a single image.

In the first application we used both the experiences in dataset creations (introduced in section 3.2) and the experience in intrinsic decomposition to define a deep architecture with physical constraints to remove shading effects from the document images. Similarly, in the second application we modified our SID2 dataset (presented in the chapter 4) for the relighting task by capturing same scene under 10 different lighting conditions and we combined intrinsic image estimation approach (used in chapter 3) with light parameter estimation (defined in chapter 4) for single image scene relighting.

## 5.1 Removing light-effects in Documents

The motivation for our first application is to remove illumination effects from document images to improve the performance of automatic content processing algorithms e.g. Optical Character Recognition (OCR). The performance of these algorithms is greatly effected by the irregular shape of the paper and the shading and shadow effects caused by diverse lighting conditions in the real world. Current fully-supervised methods can remove these effects but they require a large number of document images in versatile lighting conditions to train. Therefore, Current state of art methods opted for completely synthetic or hybrid datasets. However,

the document shadows and shading removal results still suffer because (a) prior methods used uniform color statistics which limits their applicability for more complex shape and texture documents found in real scenarios; (b) synthetic dataset uses non-realistic and simulated lighting to train the networks. In this task, we propose to handle these shortcomings with the following contributions. First, we build a large hybrid Doc3DShade dataset with physically accurate and realistic shading under complex illumination conditions, which is impossible to obtain with a rendering engine only. Second, a deep neural network based on the intrinsic image formation model (introduced in chapter 1) to estimate shading free reflectance image.

### 5.1.1 Introduction

Document digitization is the procedure to transform important and valuable paper document information into machine encoded text. This helps to retrieve, process, understand and share the document information in our everyday life. Universal presence and the increase in the usage of mobile phone devices provide a convenient and instant way to digitize documents by capturing them in the wild. These captured images are only useful if it can be easily processed for later text retrieval and content processing applications. Therefore, it is desirable to capture the image so that most of the document content is preserved. However, the interaction between the scene lighting and the shape of the documents naturally produces the shape and shading effects in the captured document images which reduces the overall performance and reliability of automatic content processing algorithms. Moreover, the presence of non-uniform multiple color light sources and diverse paper textures in the in-the-wild increases the entire complexity of the task. As an example, the part of the text is undetected in the leftmost image of figure 5.1 due to shadows and shading.

Recent deep learning based methods [53] can be used to remove shading and shadows effects from documents images to estimate clean reflectance image but these methods require large and accurate training dataset to achieve satisfactory performance. Imposing additional domain-specific physical ground-truth helps to improve the generalization performance of such models [26, 113, 115]. As discussed previously in chapter 3, creating accurate real dataset for intrinsic image decomposition is a challenging task. To remove illumination effects from the documents with deep learning models would require a large number of images with ground-truth reflectance and shading components captured in real environment with different illumination effects. To overcome this challenge, we propose a weakly-supervised

## 5.1. Removing light-effects in Documents

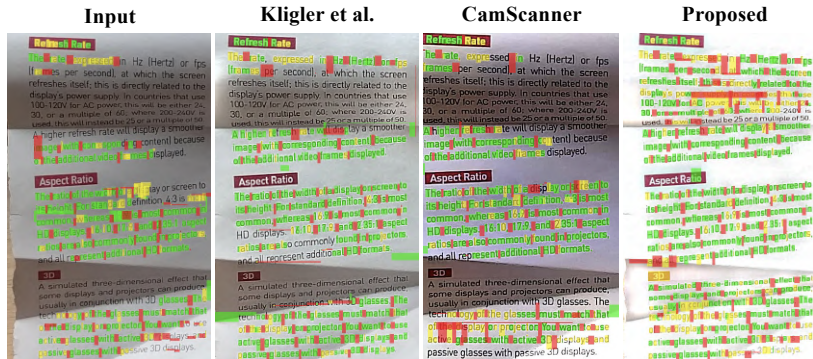


Figure 5.1: OCR accuracy (by Tesseract[88]) comparison on document images pre-processed with our proposed method vs. Kligler *et al.* (state-of-the-art document shadow removal method[62]) vs. the commercially available document capturing application CamScanner.

method to separate reflectance and shading, and train the network by using physical constraints of image formation (introduced in chapter 1). Moreover, to facilitate the training we created, Doc3DShade dataset with our hybrid dataset creation pipeline (introduced in chapter 3) that combined realistic illumination with synthetic texture in rendering engine [23].

Earlier approaches [3, 121] to remove shadow and shading effects from documents generally depends on local color priors and impose strong assumptions about uniform background color. Therefore, the applicability of these algorithms is limited to certain class of documents having minimal colors and graphics. Furthermore, few other methods assume that the shape of paper is almost flat [56, 62], which would also reduce their application for wrapped papers images captured under complex illumination conditions of the real world. To exhibit the ineffectiveness of these methods for more complex shape documents captured under non-uniform illuminations, we compare OCR results in figure 5.1. We compute OCR after pre-processing the input image (figure 5.1) with the method of Kligler *et al.* [62] (current state-of-the-art for document shadow removal), a commercially available document processing application, and our proposed reflectance estimation scheme. Our method shows a significantly higher number of detected characters proving the effectiveness of the intrinsic image-based modeling of document images over traditional approaches.



We proposed a self-supervised network for this application that has two learning-based modules, WBNet and SMTNet, which are trained on our new multi-illuminant dataset, Doc3DShade that is build on top of Doc3D [26]. We formulate the problem of document reflectance estimation based on the physics of image formation that we discussed in depth in chapter 1, i.e., an image  $I$ , is a pixel-wise product of shading  $S$  and reflectance  $R$ ,  $I = R \cdot S$  images.

Doc3Dshade is created in our lab setup with our novel combined synthetic-real dataset creation pipeline that we presented in chapter 3. In this dataset, we capture realistic shading along with its material properties  $MS$  of a deformed non-textured document under a large range of realistic colors, and diverse multi-illuminant conditions of our lab setup. Note that  $MS$  contains both the shading and the color of the paper material. We render real document textures  $T$  in Blender [23] and create  $I$  by combining  $I = T \cdot MS$ . Our formulation is similar by considering  $R = T \cdot M$ . Since the shading component is physically captured in controlled environments, these images are clearly superior in terms of physical illumination effects than the existing Doc3D [26] images. We have generated 90,000 images in Doc3DShade which effectively double the size of Doc3D.

The WBNet and SMTNet address white-balancing and shading removal respectively. WBNet inputs an RGB image and estimates a white-balanced image where the illuminant color is removed. We can train WBNet with our Doc3DShade dataset which contains a white-balanced image for every input. In the second module, SMTNet takes the white-balanced image and regresses a per-pixel shading image and paper material image. The use of the white-balanced image allows us to train SMTNet in a self-supervised manner using a physical constraint, the chromatic consistency of the white-balanced and the reflectance image. Chromatic consistency states that the white-balanced output’s chromaticity is same as true reflectance of the paper and the texture.

We thus, eliminate the costly need to physically capture explicit shading and paper reflectance ground-truth. Our method shows strong results on challenging illumination conditions and generalizes across different document types. The practical significance of the proposed method is demonstrated by a 21% decrease in OCR error rate when our shading removal is used as pre-processing for OCR.

### 5.1.2 Training Dataset: Doc3DShade

In this section, we introduce complete procedure to build Doc3DShade dataset. The dataset has large number of document images that combines diverse, realistic

## 5.1. Removing light-effects in Documents

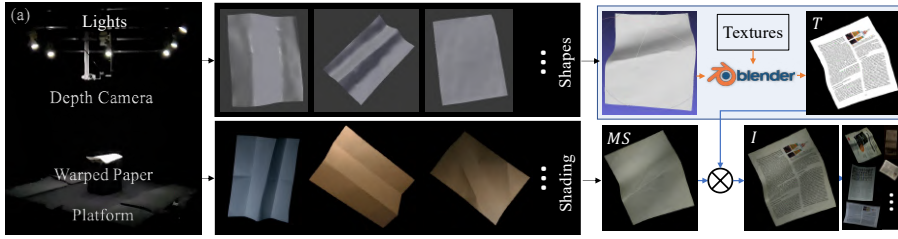


Figure 5.2: Data Creation Pipeline: (a) Shows the hardware setup. The captured shapes are textured in Blender and combined with the captured shading image by element-wise multiplication to create  $I$ . The  $\otimes$  denote the element-wise multiplication. The ‘orange’ and ‘blue’ arrows denote the rendering and the combining.

illumination scenarios with natural paper textures.

Doc3DShade increases the physical correctness of Doc3D, a recent document dataset [26]. The contributions of Doc3DShade are following;

- It captures physically accurate and realistic shading under complex illumination conditions of our lab setup (see chapter 3), which is impossible to obtain using rendering engines (which use approximate illumination models to simulate light transport).
- It contains various paper materials with different reflectance properties whereas a synthetically rendered dataset like Doc3D uses purely diffuse material to render images.

Doc3DShade is thus physically more accurate and can be used to model scene illumination in a physically grounded way.

**Capturing 3D Shape and Shading:** We modified our capturing setup (see 3.2.3) for this application by removing central light and introduced Intel RealSense RGB Depth CameraD435 on top of the scene to acquire aligned shape and shading images. (figure 5.2(a)). The new capturing setup uses only 8 directional lights. We cover tabletop and walls with black curtains to mitigate any inter-reflectance on the documents. We put a 20cm stand at the center of the rotatory table to hold the warped documents. This heightened stand enables us to segment the depth maps from the background. We make sure the paper is stable against the platform movement. It is necessary to ensure the perfect alignment of the captured images. The camera is mounted at a 50cm distance from the rotatory platform. The directional lights are at 100cm from the platform and directed towards the center of



Figure 5.3: Doc3DShade capture setup modified from our original setup (figure 3.4), We removed Central light and introduced Intel RealSense RGB Depth CameraD435 camera on top to acquire shape and shading images.

the platform. Rotation is performed with pan range of  $[0^\circ, 180^\circ]$  and tilt range of  $[-15^\circ, 15^\circ]$ . A photo of our platform is shown in Fig. 5.3.

To capture 3D shapes of documents, we randomly deformed textureless papers and placed them on the rotatory platform. The platform is randomly rotated and each document is captured with a random combination of single and multiple color directional lights. Similar to our SID family of dataset, we defined light chromaticity to be constrained around the Planckian locus [128] to simulate natural lighting conditions. We also captured each document under white lights to use it later as ground-truth for white balancing. To include various material properties in the shading images, we have used 9 diffuse paper materials that are used in various types of documents such as magazines, newspapers and printed papers, etc. In summary, we obtain the following data for each warped paper: a 3D point cloud and multiple images with shading under single and multiple lights of different color

temperatures. An illustration of the captured shapes and corresponding shading images are shown in Fig. 5.2. Note that the captured shading ( $MS$ ) map in this setup is a combined form of the paper material ( $M$ ) and the shading ( $S$ ) component.

**Image Rendering:** We create the 3D mesh for each point cloud following [45]. Each mesh is textured with a random document image and rendered with diffuse white material in Blender [23] (Fig. 5.2). In total, we have used  $\sim 5000$  textures collected from various documents such as magazines, books, flyers, etc. Each texture image is combined with a randomly selected single light shading image of the same mesh. To create more variability, we uniformly sample and linearly combine two single light shading images to simulate a fake multi-light shading image:  $I = T \cdot (aMS_1 + (1 - a)MS_2)$ , with  $a \in [0, 1]$ . Additionally, for each image, we also render the white balanced image by combining the synthetic texture image and the shading image captured under white light. We have created 90K images with diffuse texture and white-balance images as ground-truths, for training and testing.

### 5.1.3 Document Reflectance Estimation Network

Our reflectance estimation framework for document images captured under non-uniform lighting in a real-world scenario follows a two-step approach for illuminant and shading correction and leverages the physical properties of the image formation model (introduced in chapter 1). In the first step, we estimate a white-balanced image to neutralize the color of the scene illuminants. In the second step, we disentangle shading, texture, and material of the document, which allows us to obtain the shading-free image.

We used formulation of Hui et al. in [52] which proves that chromaticity of white balanced image  $C_{wb}^c$  is equal to chromaticity of reflectance component  $C_R^c$ . We used this physical constraints,  $C_{wb}^c = C_R^c$ , to estimate the intrinsic components of a document image given a single image  $I$ . Our approach is based on using two sub-networks: WBNet and SMTNet (Fig. 5.4). The first network WBNet estimates the white-balance kernel,  $\hat{WB}$ , whose chromaticity,  $C_{wb}$ , is used in the subsequent network, SMTNet, that in turn will estimate the shading,  $\hat{S}$  and the material,  $\hat{M}$  in a self-supervised fashion. We choose to employ self-supervised training since separate ground-truths are not available for  $S$  and  $M$ .

**WBNet: White-balance Kernel Estimation.**

The first sub-network is designed to estimate the per-pixel white-balance kernel  $WB$ . We treat this task as an image-to-image translation problem. Given an input color image,  $I$  the white-balance kernel,  $WB$  is estimated using a UNet [99] style encoder-decoder architecture with skip connections. The predicted white-balanced image,  $\hat{I}_{wb}$  is then estimated by multiplying  $\hat{WB}$  and  $I$ . Note that although it is possible to directly estimate the white-balanced image  $\hat{I}_{wb}$  from  $I$ , convolutional encoder-decoder architectures cannot preserve sharp details such as text. On the other hand,  $WB$  is a smooth per-pixel map that is much easier to learn in reality.

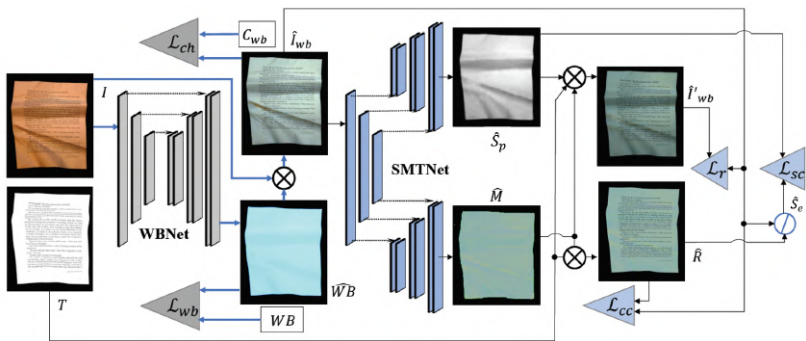


Figure 5.4: Proposed framework: The WBNet takes RGB image,  $I$  as input and produces the white-balance kernel ( $\hat{WB}$ ). The white-balanced image,  $\hat{I}_{wb}$  is then forwarded to the SMTNet which regresses the material ( $\hat{M}$ ) and the shading,  $\hat{S}_p$ . The  $\otimes$  denote the element-wise multiplication, ( $/$ ) denote division and the triangles denote the loss functions.

**Loss Function:** To train the WBNet we essentially use two loss terms, one on the predicted white-balance kernel ( $\mathcal{L}_{wb}$ ) and another on the chromaticity of the estimated white-balanced image ( $\mathcal{L}_{ch}$ ). As proved in [52], chromaticity is a shading invariant term, as is also the white-balance kernel,  $WB$ . Thus, it is intuitive to learn  $WB$  using a loss that is shading invariant. Therefore, we prefer to use the  $\mathcal{L}_{ch}$  instead of a standard reconstruction loss on the predicted white-balanced image,  $\hat{I}_{wb}$ .

Additionally, we force intensity at each pixel to be preserved after white-balancing, which acts as a constraint on the predicted image. The combined loss is:

$$\mathcal{L}_{wb} = \mathcal{L}_{wb}(\hat{WB}, WB) + \alpha_1 \mathcal{L}_{ch}(\hat{C}_{wb}, C_{wb}) + \alpha_2 \mathcal{L}_{int}(\hat{I}_{wb}, I) \quad (5.1)$$

where  $\hat{W}B$ ,  $\hat{C}_{wb}$ ,  $\hat{I}n_{wb}$  are the predicted white-balance kernel, chromaticity and intensity image of the predicted white-balanced image respectively. Corresponding ground-truths are available in our training dataset. The  $\alpha$ 's are the weights associated with each loss term. We use  $L_1$  distance for each loss term. For  $\mathcal{L}_{wb}$  and  $\mathcal{L}_{ch}$  losses we use a mask to avoid pixels where ground-truth pixel values are zero.

### SMTNet: Separating Material, Texture and Shading.

Given the estimated white-balanced image  $\hat{I}_{wb}$  the second module, SMTNet estimates the material  $\hat{M}$  and shading  $\hat{S}$ . The structure of the network is illustrated in Fig. 5.4. The network consists of one encoder and two identical decoder branches, MNet and SNet. MNet and SNet regress the  $\hat{M}$  and  $\hat{S}$  from the input image respectively. Ideally, it is possible to separately learn  $\hat{M}$  and  $\hat{S}$  in a supervised manner if the corresponding ground-truths are available. But captured shading ( $MS$ ) in our dataset is a combined representation of  $M.S$  which is further combined with the diffuse textures  $T$  to generate our training images  $I = M.T.S$ . Therefore, we resort to a self-supervised approach to train SMTNet for disentangling  $M$  and  $S$  given  $I$ , assuming  $T$  is available as ground-truth by following intrinsic image model.

**MNet:** Given the predicted material image,  $\hat{M}$  from the MNet branch, we can estimate the reflectance image as  $\hat{R} = \hat{M}.T$  and estimate the shading image as  $S_e = I/(\hat{M}.T)$ . To train this branch, we use the fact that chromaticity of reflectance image is equal to chromaticity of white balanced image [52]. If the predicted material  $\hat{M}$ , is correct the chromaticity of the  $\hat{R}$  and chromaticity of the input white-balanced image should be the same. We use this chromatic consistency loss to train the MNet branch.

**SNet:** The other branch, SNet predicts the shading image  $S_p$ . To ensure the consistency of predicted material and predicted shading,  $S_p$  should be equal to the estimated shading from the MNet branch,  $S_e$ . We use this shading consistency loss to train the SNet branch.

The SMTNet is a modified UNet architecture with a single encoder and two separate decoders. Decoders share encoded features but use different skip connections to the encoder.

**Loss Function:** The two primary loss functions used to train the SMTNet are the chromatic consistency ( $\mathcal{L}_{cc}$ ) and the shading consistency ( $\mathcal{L}_{sc}$ ) loss. Additionally, we use the reconstruction loss to ensure the predicted material image,  $\hat{M}$  and shading image,  $S_p$  reconstruct the input image when combined with the input

diffuse texture,  $T$ . We also add smoothness constraints on the predicted material and shading. Specifically, the  $L_1$  norm of the gradients for the material,  $\|\nabla \hat{M}\|$  and  $L_1$  norm of the second order gradients for the shading,  $\|\nabla^2 \hat{S}_p\|$  are added as regularizers. Whereas the first term ensures piecewise smoothness of the  $\hat{M}$ , the second term ensures a smoothly changing shading map. The combined loss function is:

$$\mathcal{L}_{smt} = \mathcal{L}_{cc}(\hat{C}_{wb}, \hat{C}_R) + \beta_1 \mathcal{L}_{sc}(\hat{S}_p, \hat{S}_e) + \beta_2 \mathcal{L}_r(\hat{I}'_{wb}, \hat{I}_{wb}) + \beta_3 \|\nabla^2 \hat{S}_p\| + \beta_4 \|\nabla \hat{M}\| \quad (5.2)$$

Where  $\hat{C}_{wb}$  and  $\hat{C}_R$  are the chromaticities of input white-balanced image and estimated reflectance image. The  $\hat{S}_p$  and  $\hat{S}_e$  are the predicted and estimated shading.  $\hat{I}'_{wb}$ ,  $\hat{I}_{wb}$  are the reconstructed and input white-balanced image. The  $\hat{I}'_{wb}$  is estimated as:  $\hat{M}.T.\hat{S}_p$ , the product of predicted material image, shading image and input diffuse texture. The  $\beta$ 's are the weights associated with each loss term. For each loss term we use the L1 loss.

We independently train each network using 80K training and 10K validation images. The training and validation sets do not contain any common 3D shapes. Both networks are trained for 100 epochs using their respective loss functions,  $\mathcal{L}_{wbn}$  and  $\mathcal{L}_{smt}$ . We use the Adam optimizer to train the network with weight decay of  $5e-4$  with an initial learning rate of  $1e-4$  for WBNNet and  $1e-3$  for SMTNet.

### 5.1.4 Evaluation and Results

For the quantitative and qualitative evaluation of our approach, we have experimented with multiple datasets that are captured under varying illumination conditions. Our evaluation contains three comparative studies. First, we show how the proposed method behave in intrinsic decomposition of document images. Second, we use the proposed approach as a post-processing step in a document unwarping pipeline [26] to show the practical applicability of our method in terms of OCR errors. Third, we show an extensive qualitative comparison with current state-of-the-art document shadow removal methods [3, 56, 62, 121]. From these experiments, we show a 21% improvement in OCR when used as a pre-processing step and also show strong qualitative performance in intrinsic document image decomposition and shadow removal tasks.

## 5.1. Removing light-effects in Documents

DewarpNet results	Image Quality		OCR Errors	
	MS-SSIM	LD	CER	WER
with shading	0.4692	8.98	0.3136	0.4010
w/o shd [26]	0.4735	8.95	0.2692	0.3582
w/o shd (Ours)	<b>0.4792</b>	<b>8.74</b>	<b>0.2453</b>	<b>0.3325</b>

Table 5.1: Quantitative comparison of [26]’s unwarping quality on DocUNet benchmark dataset [78] when our proposed approach is applied as a pre-processing step before OCR.

### Intrinsic Document Image Decomposition.

In this section we evaluate the performance of our method in decomposing intrinsic light components on two datasets e.g., DocUNet benchmark [78] and MIT Intrinsic [12]. The DocUNet benchmark contains shading and shadow-free flatbed scanned version. Practically, scanned images are the best possible way to digitize a document and can be considered as the reflectance ground-truth. For this experiment, we apply our method on the benchmark images, then use the pre-computed unwarping maps from [26] to unwarped each image which aligns each image with its scanned reflectance ground-truth. The quantitative results for image quality before and after applying our approach is reported in table 5.1, in terms of the unwarping metrics: multi-scale structural similarity (MS-SSIM) [124] and Local Distortion (LD) [131]. The qualitative results are shown in figure 5.6. Improvement of the MS-SSIM and LD are limited since these metrics are more influenced by the quality of the unwarping. Qualitative images show significant improvement over shading removal applied in [26]. It is due to explicit modeling of the paper background and illumination, which enables our model to retain a consistent background color. Few additional qualitative results on real images captured under complex illumination are shown in figure 5.5. Intermediate output of WBNet in figure 5.5 demonstrates good generalization to real scenes.

Additionally, in figure 5.8(a), we show that our method generalizes for ‘paper-like’ objects of MIT-Intrinsic [12], e.g., paper and teabag objects without any fine-tuning. Interestingly, IIW results show that it fails to preserve the text on the ‘teabag’, which proves general intrinsic methods are probably not suitable for document images and calls for further research attention to specially design intrinsic image methods for documents.



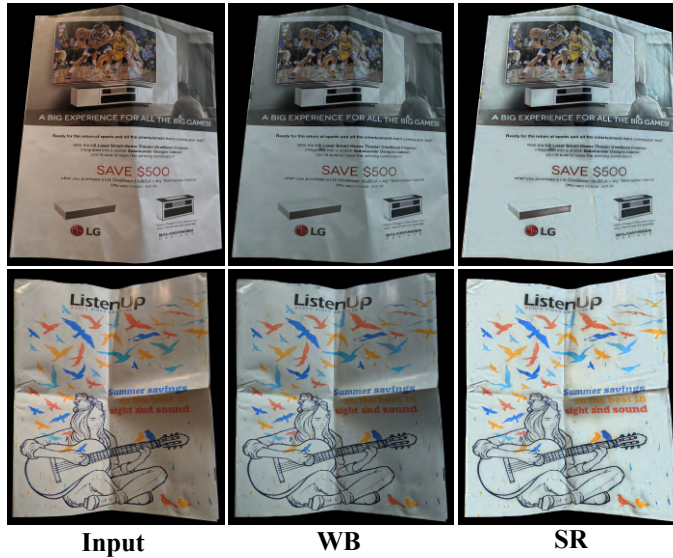


Figure 5.5: Qualitative results on real images after applying white balancing (After WB) and shading removal (After Shd. Rem.). Input images are non-uniformly illuminated with two lights.

### Pre-processing for OCR.

To demonstrate practical application of our approach, we compare OCR performance on the DocUNet benchmark images using word error rate (WER) and character error rate (CER) detailed in [26]. At first, the same unwarping step is applied as described in section 5.1.4. Then we perform OCR (Tesseract) on the DewarpNet OCR dataset before and after applying the shading removal and compare the proposed approach with the shading removal scheme presented in [26]. The results are reported in table 5.1, where we can see a clear reduction of the error in character recognition that represents a 21% performance increase.

### Shadow Removal & Non-Uniform Illumination.

Although our method is not explicitly trained for shadow removal tasks it shows competitive performance when compared to previous shadow removal approaches [3, 56, 62, 121]. We show the qualitative comparison in figure, 5.7(a), 5.7(b) and

## 5.1. Removing light-effects in Documents



Figure 5.6: Results on real-world images from [78]. [26]’s shading removal method fails to retain the background since shading, illumination and document background is modeled as a single modality.

5.7(c) on the real benchmark datasets provided in the works [3], [56], and [121] respectively. Our method consistently performs well on different examples with soft shadows and shading but fails on hard shadow cases such as the image at the bottom row of 5.7(c). Additionally, in figure 5.8(b) we show the performance of our method in challenging multi-illuminant conditions with shadows and shadings available in a recent OCR dataset [84]. We can see the results of after our white-balancing (WBNet) in the WB column of 5.8(b), and in column SMR we can see how SMTNet gracefully handles strong illumination conditions in real scenarios.

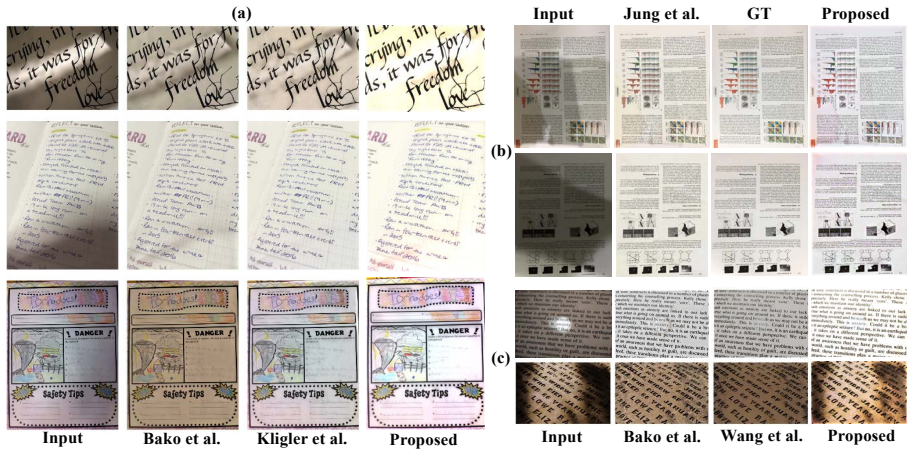


Figure 5.7: Comparison with existing shadow removal methods [3, 56, 62, 121] on real image sets: (a), (b), (c) are obtained from [3, 56, 121] respectively. These comparisons show our method well generalizes on soft shadows. We report a fail case on hard shadows at the bottom row of (c).

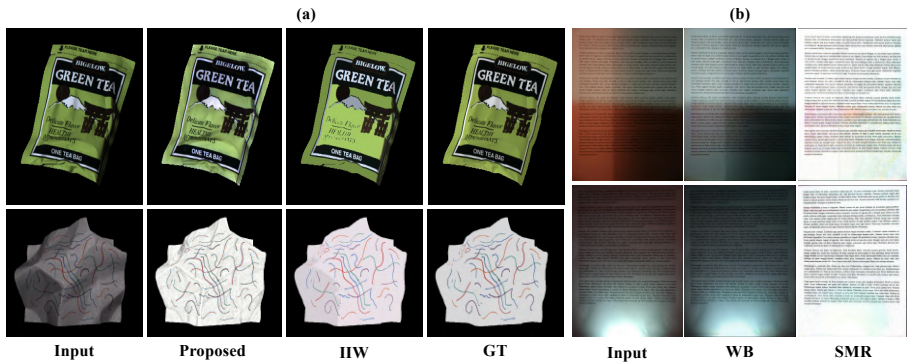


Figure 5.8: (a) Comparison with IIW method [12], IIW fails to accurately preserve text. (b) Results on multi-illuminant OCR dataset [84], WB is white-balanced image and SMR is output after removing material and shading.

## **5.2 Single Image Relighting**

In the second application, we tackled the problem of single image scene relighting. It is a highly complex task from a classical computer vision point of view, since it entangles some complex estimations at once, which are: (a) scene light direction, (b) shadow detection and removal; and (c) 3D properties of objects in the scene to be re-rendered. In the frame of deep architectures a wide range of options are emerging to face all this complexity at a time. However, different ways of approaching relighting provokes high diversity of goals, scenarios, light properties representations and datasets that we believe is not helping for a clear progress.

### **5.2.1 Introduction**

The task of image relighting is to generate a realistic new version of the original image so that it appears to be illuminated by a new given light condition. In recent years, relighting has seen to be a necessary part in a variety of fields such as Augmented Reality (AR) [90], professional photography aesthetic enhancement and photo montage [46]. Deep Neural Networks are introducing more opportunities to solve this task, since large synthetic datasets can be used for training and overcoming the inherent difficulty of the relighting problem. Even though relighting is not a new topic in computer vision [85], it is still an ongoing challenge. Recent research has proposed some deep architectures for different application scenarios. One of the most popular application has been face relighting, which works on portrait datasets and has been explored in [90, 115, 135]. Relighting of outdoor images has been addressed by Philip et al. [95] using a multi-view dataset and presenting a method based on proxy geometry. These works have obtained some competitive results on their specific applications. However, there are fewer studies about relighting from a single image of generic scenes, which is the goal of this research work.

Recently, an image relighting challenge [47] was held on the VIDIT dataset (*Virtual Image Dataset for Illumination Transfer*) [46] formed by synthetic images with very high complexity, including highly crowded scenes, with large dark areas and transparent surfaces. The challenge focus on three different approaches to the relighting problem: one-to-one relighting, estimation of illumination settings and any-to-any relighting. The best results were obtained by [29] and [122]. The first work [29] is based on an encoder-decoder scheme that resolves illumination estimation in the latent space. In the second [122], authors carry out a single image relighting through a novel Deep Relighting Network (DRN) with three parts:

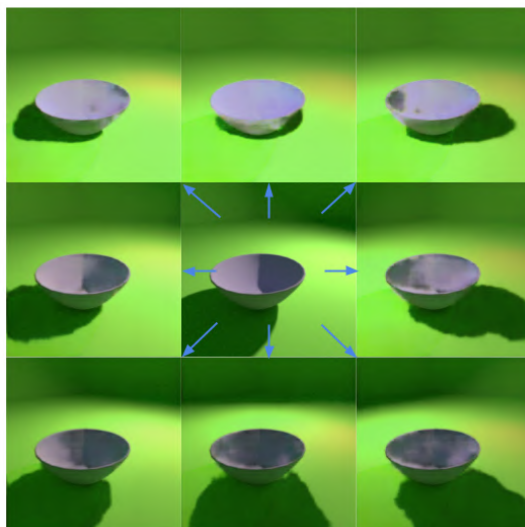


Figure 5.9: Single Image Relighting: generation of relit versions of the original image in the center for 8 different light on-top positions in front-right, front-top, front-left, left, right, back-right, back, back-left (from left to right and top to bottom).

scene reversion, shadow prior estimation and re-renderer. We will compare this second case with our proposed methods.

There are three implicit reasons that make single image relighting a difficult problem. Firstly, it requires discarding the effects caused by the original light condition, this is, removing shading effects and in particular, removing cast shadows which is one of the main obstacles to achieve proper relighting. Secondly, it needs to infer some intrinsic shape properties of the scene objects from the image shading. This difficulty particularly grows when we move from specific scenes, such as faces, to generic scenes. Thirdly, based on the target light condition and the estimated shape properties, the scene have to be rendered again, and new shadows must be reasonably derived. The aim of solving multiple complex tasks all entangled in one problem presents the perfect nature to be solved with an end-to-end deep learning architecture.

Current approaches are showing promising results but they are still far from being solved. We believe the main cause is the high diversity of goals, scenarios, representation of light properties and datasets, that impedes a clear progress. For

this task, we propose taking a step back in order to move forward. Thus, our contributions are twofold:

- We build a basic dataset of simple scenes, with a comprehensible number of object-background interactions, but with enough diversity to bring them closer to real scene complexity. The SID2 dataset (from chapter 4) is modified for this application
- We propose and test some baseline methods that goes from the simplest U-Net to architectures with multiple decoders using different levels of intrinsic components to help in the relighting task. These network architectures are designed to combine intrinsic approach of chapter 3 with light parameter estimation of chapter 4 for single image scene relighting.

### 5.2.2 SID3 Dataset

Using the methodology of SID2 dataset presented in chapter 4, we built a new dataset called SID3. We picked this option with two main criteria: (a) based on scenes to be simple enough to create a comprehensible number of object-background interactions; and (b) with some level of diversity to bring them closer to the complexity of fully textured real scenes. In this way we try to simplify the high complexity of the VIDIT scenes, ensuring physical coherence bypassing the use of environmental maps and creating sufficient shading effects to train deep learning models.

SID3 has 10,025 scenes, and each scene has 10 different light conditions, resulting in 100,250 images with intrinsic data for the whole image. The images are generated by the open source Blender rendering engine. The synthetic scenes are formed by 3D objects surrounded by walls. The 3D objects are randomly selected from various categories of the *ShapeNet* dataset [110] including electronics, pots, bus, car, chair, sofa, airplane. Roughness parameter is used to control how much light is reflected back from the object surface. Walls are set either to homogeneous colors or rich set of textured patterns are randomly selected. As a result, the dataset presents significant reflectance and shading variations across different object and background surfaces.

The created scenes are illuminated by a single oriented color source. Light properties include color and position. Light color is represented by one single parameter, that is its color temperature. For each scene, 10 images are generated using random pan and tilt light positions on a upper semi-sphere (like in figure 5.10)

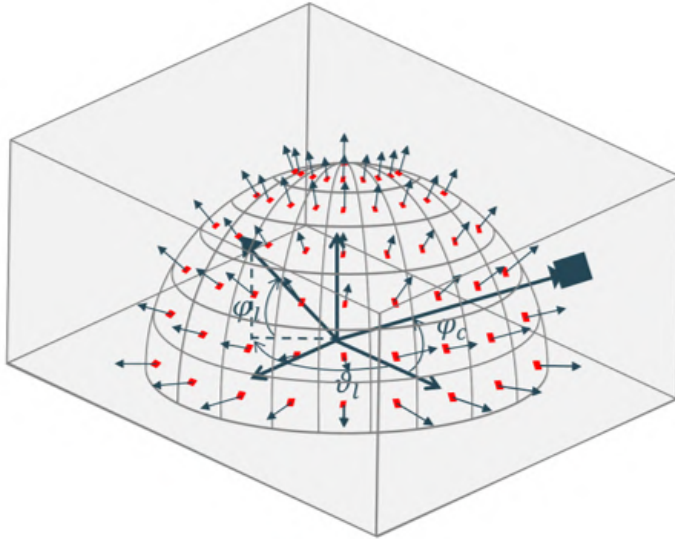


Figure 5.10: Synthetic world for image generation

whose radius can randomly vary from 20 to 50 meters. Light is always oriented to the center of the semi-sphere. Some image examples of the dataset are displayed in Fig. 5.11. The ground-truth provides the 10 corresponding shading and reflectance components. Heretofore, we will refer to the ground-truth as GT.

### 5.2.3 Network Architectures

In this section we explain different architectures we want to evaluate on single image relighting problem, after being trained on our SID3 dataset. We move from the simplest U-net (1 Encoder to 1 Decoder) architecture to different versions that increase the number of Decoders while constraining the training to the estimation of new intrinsic components. In Fig. 5.12 we plot a schema of our 3 proposed methods, that we refer to as 1-to-1 U-NET, 1-to-2 Intrinsic and 1-to-3 Intrinsic.

In these architectures we also include the estimation of the input image light properties as part of the prediction. As mentioned above, the light properties are represented by pan, tilt and color temperature. To facilitate the training, we transform pan and tilt into their cosine and sine values, and convert color temperature



## 5.2. Single Image Relighting

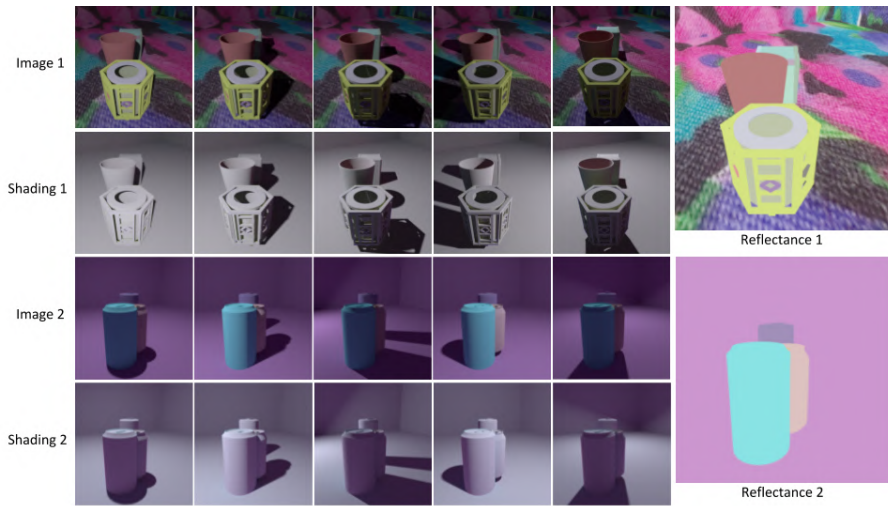


Figure 5.11: Some examples of the SID3 dataset. For 2 scenes, the reflectance component is on the right column and 5 different light conditions are shown from left to right. For each light condition we show the image (top row) and its corresponding intrinsic shading (bottom row).



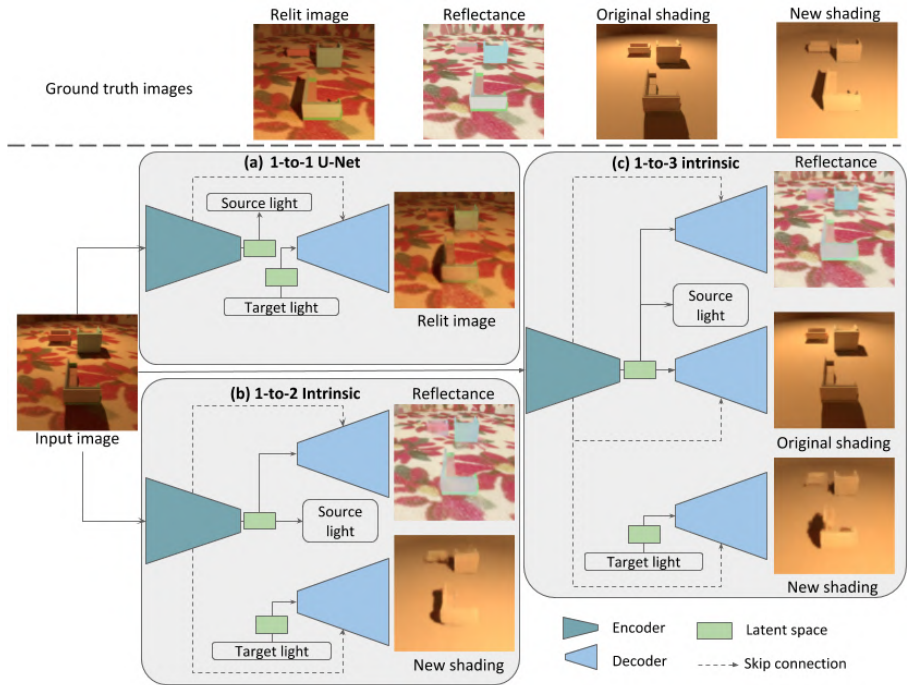


Figure 5.12: Proposed network architectures: (a) 1-to-1 U-Net with one encoder and one decoder (b) 1-to-2 intrinsic with one encoder and two decoders (c) 1-to-3 intrinsic with one encoder and three decoder

into the RGB format. In this way, their range are normalized to [0, 1] that facilitates the definition of the corresponding losses. The final formula is given by:

$$\mathbf{L}_p = [\cos(\text{pan})/2 + 0.5, \sin(\text{pan})/2 + 0.5, \cos(\text{tilt})/2 + 0.5, \sin(\text{tilt})/2 + 0.5] \quad (5.3)$$

$$\mathbf{L}_c = [R/255, G/255, B/255] \quad (5.4)$$

where  $\mathbf{L}_p$  and  $\mathbf{L}_c$  are vectors representing position and color of the image light source respectively.

### Basic 1-to-1 U-NET Architecture.

Considering the relationship between the input and relit images, the relighting task can be seen as image-to-image translation problem. We propose a basic architecture that resembles to the structure of the U-net network [99] described in pix2pix [53], which is an encoder-decoder structure with skip connections, as shown in the left top of Fig. 5.12. We modify it to introduce in the U-Net bottleneck the target light condition as an input.

The encoder is formed by a series of Convolution-BatchNorm-ReLU block. The output of this encoder is a latent space which further passes through one more convolution layer and a dense layer, yielding to the light condition of the original image. After the original light condition is yielded, a new light condition is introduced to replace it. It is processed to create a new latent space, again formed by a dense layer and a transposed convolution layer, and reshaped to the same size of the output of the encoder. The actions of the decoder is like an invert of the encoder and it takes the encoded message and the new light condition to predict a relit image.

The total loss function is defined as the sum of 3 losses as:

$$\begin{aligned} \mathcal{L}_{1to1} = & \omega_1 \mathcal{L}_{L_c}(\mathbf{L}_c, \hat{\mathbf{L}}_c) + \omega_2 \mathcal{L}_{L_p}(\mathbf{L}_p, \hat{\mathbf{L}}_p) \\ & + \omega_3 \mathcal{L}_{RnS}(RnS, R\hat{n}S) \end{aligned} \quad (5.5)$$

where  $\hat{\mathbf{L}}_p$  and  $\hat{\mathbf{L}}_c$  are denoting predictions for light position and color respectively, and  $R\hat{n}S$  is the prediction of the relit image, thus  $\mathcal{L}_{RnS}$  is the relit image loss. The different losses are combined with  $\omega_i$  weights.

### 1-to-2 Intrinsic Architecture

In this second architecture we introduce the intrinsic decomposition of chapter 3 to additionally constrain the the correct decomposition of reflectance and target shading with their GT versions. This is an earlier stage before to constraint the product of these estimated components to the target relit image.

In this case, the relighting network is given in the scheme shown in the left bottom of Fig. 5.12. The new architecture present the same encoder, but two decoders. One of them is used to predict the reflectance component, and the second one is used to yield the shading component under the target light condition. At the bottleneck, the output of the encoder is a latent representation that is transferred into the decoder to estimate the reflectance component. On the other side, the new light condition after being processing is regarded as the input of the decoder for the new shading. After the reflectance component and the new shading component are predicted, the relit image is yielded by the product given in equation 1.9.

As a result, the loss function is derived as:

$$\begin{aligned} \mathcal{L}_{1to2} = & \omega_1 \mathcal{L}_{L_c}(\mathbf{L}_c, \hat{\mathbf{L}}_c) + \omega_2 \mathcal{L}_{L_p}(\mathbf{L}_p, \hat{\mathbf{L}}_p) \\ & + \omega_3 \mathcal{L}_{RnS}(RnS, R\hat{n}S) + \omega_4 \mathcal{L}_R(R, \hat{R}) \\ & + \omega_5 \mathcal{L}_{nS}(nS, \hat{n}S) \end{aligned} \quad (5.6)$$

where  $\hat{R}$  is the reflectance prediction, and  $\hat{n}S$  is the new shading prediction, this is the shading under the target light.

### 1-to-3 Intrinsic Architecture

Furthermore, an architecture with three decoders has also been implemented as shown in the right side of Fig. 5.12. Compared with the previous schemes, it introduces one more decoder to also predict the shading of the original image. Likewise the reflectance decoder this new decoder only receives the encoded information from the encoder and has no connection with the new light. In other words, a full model for intrinsic decomposition is enclosed in this new architecture. With the output of the shading of the original light, the reconstruction of the input image can be generated by the product of reflectance and the original shading.

The estimation of the original shading requires the addition of two more losses,

which results in:

$$\begin{aligned}
\mathcal{L}_{1to3} = & \omega_1 \mathcal{L}_{L_c}(\mathbf{L}_c, \hat{\mathbf{L}}_c) + \omega_2 \mathcal{L}_{L_p}(\mathbf{L}_p, \hat{\mathbf{L}}_p) \\
& + \omega_3 \mathcal{L}_{RnS}(RnS, R\hat{n}S) + \omega_4 \mathcal{L}_R(R, \hat{R}) \\
& + \omega_5 \mathcal{L}_{nS}(nS, \hat{n}S) + \omega_6 \mathcal{L}_S(S, \hat{S}) \\
& + \omega_7 \mathcal{L}_{RS}(RS, \hat{R}S)
\end{aligned} \tag{5.7}$$

where  $\hat{S}$  is the prediction of the original shading, and  $\hat{R}S$  is the prediction of the input image from the estimated original shading.

### Implementation Details

To train on the SID3 dataset we randomly divided the dataset into three sets: 80% for training set, 5% for validation set and 15% for test. Thus, the training set has more than 80,000 images with intrinsic data. Network weight were initialized by a normal distribution. We used Adam optimizer[59] with fixed learning rate of 0.001 and batch size of 32. The inputs and outputs including images and light conditions are all normalized to [0, 1], input image size is  $256 \times 256$ , and we use L1 loss for training the network. The weights of different losses were set to 1.0.

We use the track 1 of VIDIT dataset to fine-tune our 1-to-1 U-Net network in order to compare our approach with the methods in AIM 2020 relighting challenge [47]. The track 1 has 300 pairs of input images and relit images in the training dataset, and 45 pairs in the validation dataset. Since we do not have the ground-truth of the testing dataset, we use the validation dataset as the testing dataset, and employ the training dataset to both train and validate our network. In order to fit our network, the images are resized to  $256 \times 256$ .

The metrics that we used to compare the prediction and the GT image include Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) [123, 124], Learned Perceptual Image Patch Similarity (LPIPS) [133], and Mean Perceptual Score (MPS) [47].

## 5.2.4 Results and Discussion

labelsec:results In the first part of this section, we illustrate the results based on our SID3 dataset with both qualitative evaluation and quantitative evaluation. After we get the trained network, we fine-tune it on the VIDIT dataset and compare our results with other references. At last, we test our networks on some real images to

see whether they can be generalized.

### Results on SID3 Dataset

**Qualitative Evaluation.** The qualitative results of the testing dataset are shown in Fig. 5.13 and 5.14.

In Fig. 5.13 we show a full example to visualize the behaviour of the studied architectures. The first row in Fig. 5.13 is the GT, and the rest of rows are the predictions from the 3 architectures. For some methods that do not predict certain components we have blank slots. The input image is shown in the first column (from left to right) of the first row. And the first column of 1-to-3 Intrinsic present the reconstruction calculated by the product of the reflectance component and the predicted original shading component, which is our prediction of the input image. From the estimation of the reflectance component (column 2), it can be seen that 1-to-2 Intrinsic and 1-to-3 Intrinsic all make a good estimation. Only 1-to-3 Intrinsic predicts the original shading (column 3), and the prediction is quite close to the ground-truth. In the column 4, the 1-to-2 Intrinsic and 1-to-3 Intrinsic give the prediction of new shading, and we can see that the result of 1-to-3 Intrinsic is the best one. In the right column, we can see the results of the relit images for each architecture. It can be shown that the relit image of 1-to-3 Intrinsic is the most realistic and closest to the ground-truth. The shape of the cast shadow is a bit distorted 1-to-1 U-Net, and it present some artificial effects in 1-to-2 Intrinsic.

In Fig. 5.14 we show two more complex cases than the previous one. In these cases the angular distance between the source light position and target light position is larger. In the first case, the input image presents a scene illuminated from the left, and the objective relit image is to be illuminated from top-right. This is around 135 degrees in pan. As it is shown in the results, our model almost remove the original cast shadows and generate the new cast shadow. In addition, the image quality obtained by 1-to-3 Intrinsic is in general better than the other two. In the second case, the light direction of the input image is from the back while the the new light is from the top-right. If we compare the results from 1-to-1 U-Net, 1-to-2 Intrinsic and 1-to-3 Intrinsic, all of them remove the shadow at the back of the sofa, but the 1-to-3 Intrinsic seems to be the one that better renders the new shadow if we compare with the ground-truth. Another advantage of 1-to-2 Intrinsic and 1-to-3 Intrinsic over 1-to-1 U-Net is the presence of less noise. In general, we can say that the intrinsic decomposition really help the relighting according to the qualitative results we observe.

## 5.2. Single Image Relighting

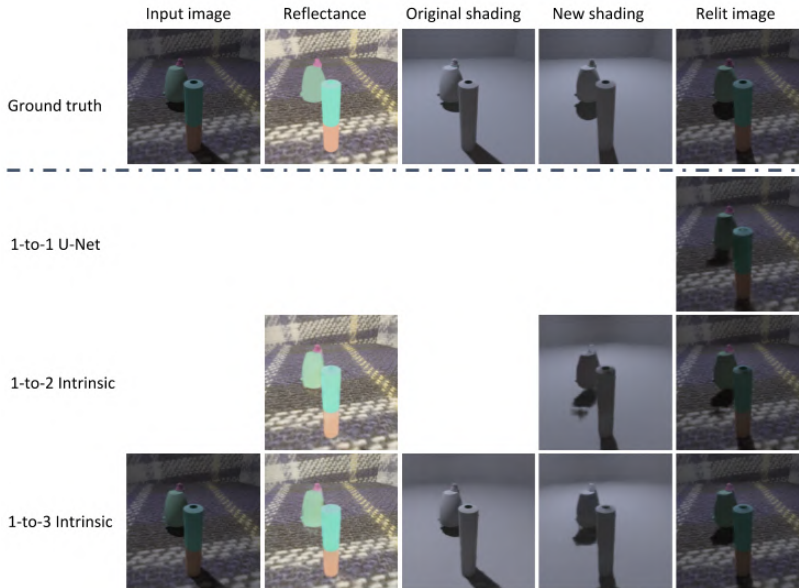


Figure 5.13: A full example of the results on SID3 dataset with all the intrinsic components.

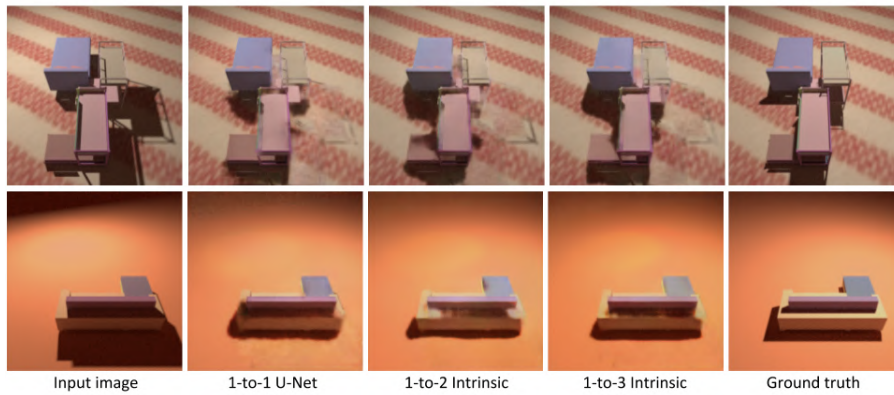


Figure 5.14: Two complex examples on SID3 dataset.

**Quantitative Evaluation** We tested the network quantitatively on the testing dataset with more than 1500 scenes. The results of the relit images output by the network are obtained with 4 different metrics, which are compiled in table 5.2. According to the table, the 1-to-3 intrinsic network gets the best results in all the metrics. And 1-to-2 intrinsic network has obvious advantages over 1-to-1 U-net when the metrics are SSIM, LPIPS and MPS. But the PSNR of 1-to-2 intrinsic network is a little less than that of 1-to-1 basic U-net. Comparing the 1-to-1 U-net and the 1-to-3 intrinsic network, the improvements are 0.11 in PSNR, 0.65 percent in SSIM, 0.92 percent in LPIPS and 0.79 percent in MPS. As a result, these make us to conclude again that the intrinsic decomposition is helping in the relighting task.

Experiments	PSNR	SSIM	LPIPS	MPS
1-to-1 U-Net	24.09	0.8986	0.1289	0.8849
1-to-2 Intrinsic	24.05	0.9025	0.1224	0.8901
1-to-3 Intrinsic	<b>24.20</b>	<b>0.9051</b>	<b>0.1197</b>	<b>0.8927</b>

Table 5.2: Quantitative results of SID3 Dataset

### Results on VIDIT Dataset

VIDIT dataset does not provide the intrinsic components of reflectance and shading for each image, so for this dataset the only possibility was to fine-tune with our 1-to-1 U-NET architecture. Table 5.3 shows the quantitative comparison with other methods on similar metrics. Our method gets the best result on LPIPS, PSNR and MPS metrics and get comparable numbers on SSIM which are state of art methods. In Fig. 5.15 we showed the qualitative comparison with some of the methods we compared before. Our method, as the rest still have to improve the shadow removal but is the only one that creates some new shadows on the correct direction according to the new light position. These results confirm the generalization capabilities of our network and the effectiveness of training on our synthetic dataset that shows some generalization on this VIDIT dataset.

### Results on Real Images

Finally, we test our trained networks (only trained by the SID3 dataset) on some real images to explore the generalization capabilities of our network. Results are shown in Fig. 5.16. The image in the first row is from the IIW dataset [13], and the

Experiments	MPS	SSIM	LPIPS	PSNR
CET_SP[97]	0.6452	0.6310	0.3405	17.0717
CET_CVLAB[97]	0.6451	<b>0.6362</b>	0.3460	16.8927
Lyl[47]	0.6436	0.6301	0.3430	16.6801
YorkU[47]	0.6216	0.6091	0.3659	16.8196
IPCV_IITM[47]	0.5897	0.5298	0.3505	17.0594
DeepRelight[122]	0.5892	0.5928	0.4144	17.4252
Hertz[47]	0.5339	0.5666	0.4989	16.9234
Image Lab[47]	0.3746	0.3769	0.6278	16.8949
1-to-1 U-Net	<b>0.6484</b>	0.6266	<b>0.3299</b>	<b>18.7803</b>

Table 5.3: Quantitative Results of VIDIT dataset

one in the second row is taken by ourselves. The relit images are predicted by two methods 1-to-1 U-Net and 1-to-3 Intrinsic.

In the first example, the input image is illuminated from the right side, and the new light condition is from the front top. We can see that our architectures are removing the shadow besides the sofa, and new shadows are generated under the table. The prediction of 1-to-3 Intrinsic seems to be more reasonable than that of 1-to-1 U-net. In the second example, the light comes from the left top, and the new light is illuminated from the back of the water-bottle. Both two methods try to generate the new shadow, and again the shadow produced by 1-to-3 Intrinsic is more complete. Thus, the intrinsic decomposition again seems to be helpful in the relighting tasks when we infer on real images different from the used dataset. In both cases we are still far from a having a good result, but it seems that strategy on training on simple scene but with strong effects can be a good one.

## 5.3 Conclusion

In chapter 5, we proved the plausibility of using intrinsic decomposition for two different computer vision applications: (a) Removing light effect in document images; and (b) Relighting a scene from a single image. In case of these practical applications we used our experience in creating the datasets and designing deep neural network to accommodate physical constraints of intrinsic decomposing



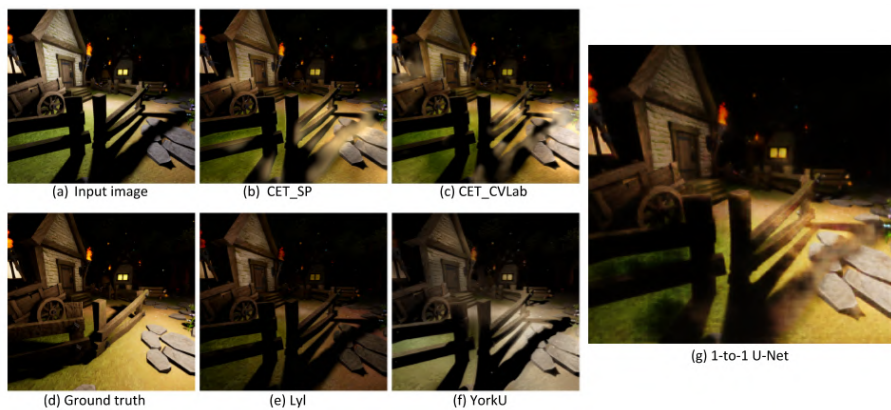


Figure 5.15: Qualitative results on VIDIT dataset, images from (a)-(f) are obtained from [47], (g) our prediction with 1-to-1 U-Net



Figure 5.16: Qualitative results on real images

model.

In the first application, we made a dataset named Doc3DShade that is based on our combined synthetic-real dataset creation methodology of chapter 3 to enhance results for automatic document content processing which are mostly affected by artifacts caused by the shape of the paper, and non-uniform and diverse color of lighting conditions. We collected a large-scale Document dataset that combines diverse, realistic illumination scenarios with natural paper textures. To remove unwanted lighting effects in this application, we designed a two-stage self-supervised neural network. The first stage neural network learns white balance document image and the second stage neural network predicts required shading free reflectance image. We designed this two-stage self supervised neural network to incorporate these physical constrains: (a) the chromaticity of the white balanced image is equal to chromaticity of reflectance image (b) physical intrinsic image model.

In the second application, we tackled the problem of single image scene relighting. To achieve this objective, firstly, we build a large dataset of surreal scenes with consistent light conditions, a comprehensible number of objects and a sufficient level of diversity to approach the problem step by step. Secondly, establish the performance of three deep architectures from a basic encoder-decoder, to some extensions that introduce physical constraints derived from the intrinsic decomposition model. Lastly, We show that our approach is showing accurate shadow removal and re-rendering in simple images, but we also show that with a proper fine-tuning our method is able to generalize to more complex dataset showing comparable or even better results of the current state of the art.



## 6 Conclusions and Future Work

Scene lighting plays a critical role in the final appearance of digital photographs and disentangling this lighting is helpful in different computer vision applications to remove lighting effects from images or to manipulate the lighting to create new effects in already captured images. The problem of decomposing images to estimate light-dependent properties, such as *shading* and *illumination*, along with material-dependent properties, such as *reflectance* or *albedo* is known as intrinsic image decomposition in the computer vision community and that is the focus of this Ph.D. thesis. We explored the use of regression-based convolution neural networks (CNN) for this challenging task and to use it for related applications. The performance of CNN depends greatly on the availability of accurate and large datasets. We identified this problem in the earlier stage of research that there were very few datasets available at that time with enough training data with versatile lighting effects to resolve this problem with deep neural networks. Creating a ground-truth dataset for this task itself is a challenging task. We have introduced novel contributions in resolving these problems for this task by following four-step methodology: (a) identify the problem, (b) create the ground-truth (c) train a deep neural network to solve the problem, (d) quantitatively and qualitatively evaluate the performance of the trained network on different ground-truth images. We already implied multiple conclusions separately in each chapter based on these contributions. In the first part of this chapter, we review these conclusions in one place, and in the second part of this chapter, we discuss possible future directions and new research lines related to this thesis.

### 6.1 Conclusion

In 1, we gave an introduction to the problem of intrinsic image decomposition in computer vision and how this has been mathematically modelled in computer vision. We also discussed the challenges and applications associated with this problem, which were driving force for this research work.

In chapter 2 we gave an overview of existing datasets and techniques along with the evaluation matrices for intrinsic image decomposition. Firstly, we briefly discuss each of the available intrinsic image datasets and analyze these datasets according to several properties. Secondly, we revised traditional and deep learning based models that have been developed to predict intrinsic components, we also provided a comparison of traditional and deep learning based models according to different parameters such as type of inputs, physical cues and type of neural networks authors used in their research work and output intrinsic components they estimate. The analysis of these detests and methods is one of the contribution of this thesis.

In chapter 3, we focused on creating large ground-truth datasets for intrinsic image decomposition in reflectance and shading with the estimation of these properties using deep neural networks from a single image. First part of this chapter described in detail different possible ways to create ground truth for this task with their pros and cons. First, we proposed a color-based data augmentation technique that extends the training data by increasing the variability of chromaticity and preserving the reflectance geometry of the ground truth. In this way, the lack of data can be partially solved with data augmentation. Secondly, We presented a pipeline to build a dataset by registering synthetic with acquired scenes to be able to automatize the process of building the dataset ground-truth. This is a really hard task that requires geometric and photometric calibration between two worlds. We used this hybrid ground-truth creation technique to create a Doc3Dshade dataset in chapter 5 to remove illumination effects from document images. Lastly, we introduced a completely synthetic dataset named SID for *Surreal Intrinsic Dataset*. This dataset has a lot of variations of shading and reflectance effects.

In second part of this chapter, we propose a versatile framework to define and train a convolutional network able to perform an intrinsic decomposition through training on a dataset with a large variety of light effects and color reflectances. our proposed CNN architecture has been defined in a simplistic way to reduce its number of parameters and enough flexible to be adapted to multiple types of visual tasks related to light effect estimation. The results obtained by all the experiments

we report in this chapter, make us to be optimistic about the capabilities of the presented approach to train networks devoted to solve task related to the estimation of light effects. In all the reported experiments we show a performance close to the state of the art of the problem of intrinsic decomposition in shading and reflectance.

In chapter 4, we focused on another intrinsic image component by estimating light source properties from a single image. We presented a method to estimate the direction and color of a scene light source from a single image. In this work we extended SID1 dataset of chapter 3 by introducing multiple objects in multi illumination scenario, with the aim of increasing the number of strong light effects and interactions to help in this task. The dataset is formed by 45,000 synthetic images with the corresponding ground truth data: direction and color of the light source. The new dataset is called SID2 dataset. In addition to the new dataset, we defined a deep architecture trained on the SID2 dataset to estimate direction and color of the scene light source. Apart from showing a good performance on synthetic images, we additionally propose a preliminary procedure to obtain light positions of the Multi-Illumination dataset[86], and, in this way, we also proved that our trained model achieves a good performance when it is applied to real scenes.

In chapter 5, we proved the plausibility of using intrinsic decomposition for two different computer vision applications: (a) Removing light effect in document images; and (b) Relighting a scene from a single image. In case of these practical applications we used our experience in creating the datasets and designing deep neural network to accommodate physical constrains of intrinsic decomposing model.

In the first application, we made a dataset named Doc3DShade that is based on our combined synthetic-real dataset creation methodology of chapter 3 to enhance results for automatic document content processing algorithms which are mostly affected by artifacts caused by the shape of the paper, and non-uniform and diverse color of lighting conditions. We collected a large-scale document dataset that combines diverse, realistic illumination scenarios with natural paper textures. To remove unwanted lighting effects in this application, we designed a two-stage self-supervised neural network. The first stage neural network learns white balance document image and the second stage neural network predicts required shading free reflectance image. We designed this two-stage self supervised neural network to incorporate following physical constrains: (a) the chromaticity of the white balanced image is equal to chromaticity of reflectance image (b) physical intrinsic image model i.e  $I = R.S$

In the second application, we combined our ideas of intrinsic decomposition

in reflectance and shading of chapter 3 and light source properties estimation of chapter 4 for a single image relighting task. Single image relighting is a highly complex task from a classical computer vision point of view, since it entangles some complex estimations at once, which are: (a) scene light direction, (b) shadow detection and removal; and (c) 3D properties of objects in the scene to be re-rendered. In order to train deep neural network to estimate light properties, we modified the SID2 dataset of chapter 4 to build a large dataset with consistent light conditions, a comprehensible number of objects, and a sufficient level of diversity to approach this problem. We established the performance of three deep architectures from a basic encoder-decoder, to some extensions that introduce physical constraints derived from the intrinsic decomposition model. We showed that our approach is showing accurate shadow removal and re-rendering in simple images.

### 6.2 Future work

The impressive performance of CNN based methods for intrinsic image decomposition and its related applications is enforcing researchers to build large ground-truth datasets with accurate and versatile illumination conditions. In this thesis, we also followed the same path and introduced novel contributions in this domain. These contributions have opened some new research lines with future research goals. In this section, we talk about these future research directions including some ideas to improve and extend our work.

The review of recent deep learning based methods and datasets for intrinsic image decomposition in chapter 2 have shown us that the recent trend in this field is to use inverse rendering based models which tries to learn reflectance and shading decomposition with other intrinsic components such as normal, shape, and lighting environmental map. In the future, we are interested to modify our dataset creation pipelines of chapter 3 to make ground-truth images for these other intrinsic components. Our surreal intrinsic dataset creation pipeline can be easily adapted to create ground-truth images for these other intrinsic components.

Moreover, we acknowledge that the combined synthetic-real dataset presented in chapter 3 still needs improvements to better replicate real-world shape and illumination conditions. In the future, we are interested to use photometric stereo and multi-view stereo algorithms to improve these discrepancies. We already presented some initial results for these algorithms in the same chapter. In the case of our synthetic datasets, the future goal should be to bring more naturalism to the images

and to increase the complexity and variation of lighting and object textures. One possible direction to explore would be to use physically based rendering materials [94] in our synthetic dataset.

Furthermore, we believe that recent adversarial neural networks [53] could help in improving the visual quality of our predicted images both in intrinsic decomposition and image relighting tasks. In addition to that, domain adaptation based methods [37] could also help to reduce the gap between synthetic and real-world scenes to help networks to generalize and perform better in real scenarios.

Finally, the presented research work in this thesis should be explored for other related computer vision applications such as shadow removal, face relighting, material re-rendering, and inverse re-rendering.

## 6.3 List of Publications

### 6.3.1 Journals

- **Hassan A. Sial**, Ramon Baldrich, and Maria Vanrell. "Deep intrinsic decomposition trained on surreal scenes yet with realistic light effects." *Journal of the Optical Society of America A*, 37.1 (2020): 1-15.

### 6.3.2 International Conferences

- **Hassan A.Sial**, Sergio Sancho, Ramon Baldrich, Robert Benavente, and Maria Vanrell. "Color-based data augmentation for reflectance estimation." In *Color and Imaging Conference*, 2018.
- **Hassan A.Sial**, Ramon Baldrich, Maria Vanrell, and Dimitris Samaras. "Light Direction and Color Estimation from Single Image with Deep Regression." In *London Imaging Meeting*, 2020.
- Sagnik Das, **Hassan A. Sial**, Ke Ma, Ramón Baldrich, Maria Vanrell and Dimitris Samaras "Intrinsic Decomposition of Document Images In-the-Wild " In *British Machine Vision Conference*, 2020.
- Yixiong Yang, **Hassan A. Sial**, Ramón Baldrich, and Maria Vanrell "Single Image Relighting Using Multi-scale cGAN with Intrinsic Constraints "**Submitted** In *British Machine Vision Conference*, 2021.





# Bibliography

- [1] K. Agusanto, L. Li, Z. Chuangui, and W.S. Ng. Photorealistic rendering for augmented reality using environment illumination. ISMAR, pages 208–216, 2003.
- [2] I. Arief, S. McCallum, and J.Y. Hardeberg. Realtime estimation of illumination direction for augmented reality on mobile devices. In CIC, volume 2012, 2012.
- [3] Steve Bako, Soheil Darabi, Eli Shechtman, Jue Wang, Kalyan Sunkavalli, and Pradeep Sen. Removing shadows from images of documents. In Asian Conference on Computer Vision, pages 173–183. Springer, 2016.
- [4] J. T. Barron and J. Malik. Intrinsic scene properties from a single rgb-d image. In 2013 IEEE Conference on Computer Vision and Pattern Recognition, pages 17–24, June 2013.
- [5] Jonathan T. Barron and Jitendra Malik. Shape, illumination, and reflectance from shading. IEEE Transactions on Pattern Analysis and Machine Intelligence, 37(8):1670–1687, 2015.
- [6] Harry Barrow and J Tenenbaum. Recovering intrinsic scene characteristics. Comput. Vis. Syst, 2, 1978.
- [7] A. S. Baslamisli, T. T. Groenestege, P. Das, H. A. Le, S. Karaoglu, and T. Gevers. Joint learning of intrinsic images and semantic segmentation. In European Conference on Computer Vision, 2018.
- [8] Anil S. Baslamisli, Hoang-An Le, and Theo Gevers. CNN based learning using reflection and retinex models for intrinsic image decomposition. In Computer Vision and Pattern Recognition, 2018.

- [9] Anil S Baslamisli, Yang Liu, Sezer Karaoglu, and Theo Gevers. Physics-based shading reconstruction for intrinsic image decomposition. Computer Vision and Image Understanding, 205:103183, 2021.
- [10] Shida Beigpour, Mai Lan Ha, Sven Kunz, Andreas Kolb, and Volker Blanz. Multi-view multi-illuminant intrinsic dataset. In BMVC, 2016.
- [11] Shida Beigpour and Joost Van De Weijer. Object recoloring based on intrinsic image estimation. In 2011 International Conference on Computer Vision, pages 327–334. IEEE, 2011.
- [12] Sean Bell, Kavita Bala, and Noah Snavely. Intrinsic images in the wild. ACM Trans. on Graphics (SIGGRAPH), 33(4), 2014.
- [13] Sean Bell, Kavita Bala, and Noah Snavely. Intrinsic images in the wild. ACM Transactions on Graphics (TOG), 33(4):1–12, 2014.
- [14] Sai Bi, Xiaoguang Han, and Yizhou Yu. An l1 image transform for edge-preserving smoothing and scene-level intrinsic decomposition. ACM Transactions on Graphics, 34:78, 08 2015.
- [15] Adrien Bousseau, Sylvain Paris, and Frédo Durand. User-assisted intrinsic images. In ACM SIGGRAPH Asia 2009 papers, pages 1–10. 2009.
- [16] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a siamese time delay neural network. IJPRAI, 7:669–688, 1993.
- [17] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, European Conf. on Computer Vision (ECCV), Part IV, LNCS 7577, pages 611–625. Springer-Verlag, October 2012.
- [18] Jim Chandler and John Fryer. Autodesk 123d catch: how accurate is it. Geomatics world, 2(21):28–30, 2013.
- [19] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. CoRR, abs/1512.03012, 2015.
- [20] Qifeng Chen and Vladlen Koltun. A simple model for intrinsic image decomposition with depth cues. In Computer Vision (ICCV), 2013 IEEE International Conference on, pages 241–248. IEEE, 2013.

- [21] D. Cheng, D.K. Prasad, and M.S. Brown. Illuminant estimation for color constancy: why spatial-domain methods work and the role of the color distribution. JOSA A, 31(5):1049–1058, 2014.
- [22] François Chollet et al. Keras. <https://github.com/fchollet/keras>, 2015.
- [23] Blender Online Community. Blender - a 3D modelling and rendering package. [www.blender.org](http://www.blender.org).
- [24] Corel datasets. <http://www.emsps.com/photocd/corelcds.htm>.
- [25] P. Das, A.S. Baslamisli, Y. Liu, S. Karaoglu, and T. Gevers. Color constancy by gans: An experimental survey, 2018.
- [26] Sagnik Das, Ke Ma, Zhixin Shu, Dimitris Samaras, and Roy Shilkrot. Dewarpnet: Single-image document unwarping with stacked 3d and 2d regression networks. In The IEEE International Conference on Computer Vision (ICCV), October 2019.
- [27] Sagnik Das, Hassan Ahmed Sial, Ke Ma, Ramon Baldrich, Maria Vanrell, and Dimitris Samaras. Intrinsic decomposition of document images in-the-wild. arXiv preprint arXiv:2011.14447, 2020.
- [28] P.E. Debevec. Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography. In SIGGRAPH, 1998.
- [29] Alexandre Pierre Dherse, Martin Nicolas Everaert, and Jakub Jan Gwizdala. Scene relighting with illumination estimation in the latent space on an encoder-decoder scheme. arXiv preprint arXiv:2006.02333, 2020.
- [30] Sylvain Duchêne, Clement Riant, Gaurav Chaurasia, Jorge Lopez-Moreno, Pierre-Yves Laffont, Stefan Popov, Adrien Bousseau, and George Drettakis. Multi-view intrinsic images of outdoors scenes with an application to relighting. ACM Transactions on Graphics, page 16, 2015.
- [31] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In NIPS, 2014.
- [32] Qingnan Fan, Jiaolong Yang, Gang Hua, Baoquan Chen, and David P. Wipf. Revisiting deep intrinsic image decompositions. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8944–8952, 2018.

## Bibliography

---

- [33] G.D. Finlayson, S.D. Hordley, C. Lu, and M.S. Drew. On the removal of shadows from images. PAMI, 28(1):59–68, 2006.
- [34] D.H. Foster. Color constancy. Vision Research, 51(7):674 – 700, 2011.
- [35] Brian Funt, Mark Drew, and Michael Brockington. Recovering shading from color images. In European Conference on Computer Vision, pages 124–132, 1992.
- [36] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multi-view stereopsis (pmvs). In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 2, 2007.
- [37] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In International conference on machine learning, pages 1180–1189. PMLR, 2015.
- [38] Elena Garces, Adolfo Munoz, Jorge Lopez-Moreno, and Diego Gutierrez. Intrinsic images by clustering. Comput. Graph. Forum, 31(4):1415–1424, June 2012.
- [39] M. Gardner, Y.H. Geoffroy, K. Sunkavalli, C. Gagné, and J. Lalonde. Deep parametric indoor lighting estimation. In ICCV, pages 7175–7183, 2019.
- [40] M. Gardner, K. Sunkavalli, E. Yumer, X. Shen, E. Gambaretto, C. Gagné, and J. Lalonde. Learning to predict indoor illumination from a single image. arXiv preprint arXiv:1704.00090, 2017.
- [41] Peter V. Gehler, Carsten Rother, Martin Kiefel, Lumin Zhang, and Bernhard Schölkopf. Recovering intrinsic images with a global sparsity prior on reflectance. In Neural Information Processing Systems, pages 765–773, 2011.
- [42] A. Gijsenij, T. Gevers, and J.V. Weijer. Computational color constancy: Survey and experiments. PAMI, 20(9):2475–2489, 2011.
- [43] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In NIPS, 2014.
- [44] Roger Grosse, Micah K. Johnson, Edward H. Adelson, and William T. Freeman. Ground-truth dataset and baseline evaluations for intrinsic image algorithms. In International Conference on Computer Vision, pages 2335–2342, 2009.

- [45] B. Haefner, Y. Quéau, T. Möllenhoff, and D. Cremers. Fight ill-posedness with ill-posedness: Single-shot variational depth super-resolution from shading. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
- [46] Majed El Helou, Ruofan Zhou, Johan Barthas, and Sabine Süsstrunk. Vedit: Virtual image dataset for illumination transfer. arXiv preprint arXiv:2005.05460, 2020.
- [47] Majed El Helou, Ruofan Zhou, Sabine Süsstrunk, Radu Timofte, Mahmoud Afifi, Michael S Brown, Kele Xu, Hengxing Cai, Yuzhong Liu, Li-Wen Wang, et al. Aim 2020: Scene relighting and illumination estimation challenge. arXiv preprint arXiv:2009.12798, 2020.
- [48] Paul Henderson and Vittorio Ferrari. Learning single-image 3d reconstruction by generative modelling of shape, pose and shading. International Journal of Computer Vision, pages 1–20, 2019.
- [49] Yannick Hold-Geoffroy, Kalyan Sunkavalli, Sunil Hadap, Emiliano Gamberetto, and Jean-François Lalonde. Deep outdoor illumination estimation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 7312–7321, 2017.
- [50] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In Proceedings of the IEEE International Conference on Computer Vision, pages 1501–1510, 2017.
- [51] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In Proceedings of the European conference on computer vision (ECCV), pages 172–189, 2018.
- [52] Zhuo Hui, Aswin C. Sankaranarayanan, Kalyan Sunkavalli, and Sunil Hadap. White balance under mixed illumination using flash photography. In IEEE Intl. Conf. Computational Photography (ICCP), 2016.
- [53] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1125–1134, 2017.
- [54] Junho Jeon, Sunghyun Cho, Xin Tong, and Seungyong Lee. Intrinsic image decomposition using structure-texture separation and surface normals. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors,

## Bibliography

---

- Computer Vision – ECCV 2014, pages 218–233. Springer International Publishing, 2014.
- [55] J.F.Lalonde, A.A.Efros, and S.G.Narasimhan. Estimating natural illumination from a single outdoor image. ICCV, pages 183–190, 2009.
- [56] Seungjun Jung, Muhammad Abul Hasan, and Changick Kim. Water-filling: An efficient algorithm for digitized document shadow removal. In Asian Conference on Computer Vision, pages 398–414. Springer, 2018.
- [57] H. Kaiming, Z. Xiangyu, R. Shaoqing, and S. Jian. Deep residual learning for image recognition. CVPR, pages 770–778, 2016.
- [58] Seungryong Kim, Kihong Park, Kwanghoon Sohn, and Stephen Lin. Unified depth prediction and intrinsic image decomposition from a single image via joint convolutional neural fields. In ECCV, 2016.
- [59] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.
- [60] D.P. Kingma and J. Ba. Adam: A method for stochastic optimization. CoRR, abs/1412.6980, 2014.
- [61] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. Optimization by simulated annealing. science, 220(4598):671–680, 1983.
- [62] Netanel Kligler, Sagi Katz, and Ayellet Tal. Document enhancement using visibility detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2374–2382, 2018.
- [63] Gudrun J Klinker, Steven A Shafer, and Takeo Kanade. The measurement of highlights in color images. International Journal of Computer Vision, 2(1):7–32, 1988.
- [64] Gudrun J Klinker, Steven A Shafer, and Takeo Kanade. A physical approach to color image understanding. International Journal of Computer Vision, 4(1):7–38, 1990.
- [65] Tomáš Krajiník, Jan Blažíček, and Joao M Santos. Visual road following using intrinsic images. In 2015 European Conference on Mobile Robots (ECMR), pages 1–6. IEEE, 2015.
- [66] P. Laffont and J. Bazin. Intrinsic decomposition of image sequences from local temporal variations. In 2015 IEEE International Conference on Computer Vision (ICCV), pages 433–441, Dec 2015.

- [67] Edwin H. Land and John McCann. Lightness and retinex theory. Journal of the Optical Society of America, 61(1):1–11, 1971.
- [68] K. J. Lee, Q. Zhao, X. Tong, M. Gong, S. Izadi, S. U. Lee, P. Tan, and S. Lin. Estimation of intrinsic image sequences from image depth video. In European Conference on Computer Vision, pages 327–340, 2012.
- [69] Louis Lettry, Kenneth Vanhoey, and Luc Van Gool. Darn: A deep adversarial residual network for intrinsic image decomposition. 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pages 1359–1367, 2018.
- [70] Wenbin Li, Sajad Saeedi, John McCormac, Ronald Clark, Dimos Tzoumanikas, Qing Ye, Yuzhong Huang, Rui Tang, and Stefan Leutenegger. Interiornet: Mega-scale multi-sensor photo-realistic indoor scenes dataset. In British Machine Vision Conference (BMVC), 2018.
- [71] Zhengqi Li and Noah Snavely. Cgintrinsics: Better intrinsic image decomposition through physically-based rendering. In European Conference on Computer Vision (ECCV), 2018.
- [72] Zhengqi Li and Noah Snavely. Learning intrinsic image decomposition from watching the world. In 2018 CVPR, pages 9039–9048, 2018.
- [73] Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and svbrdf from a single image. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2475–2484, 2020.
- [74] Fayao Liu, Chunhua Shen, and Guosheng Lin. Deep convolutional neural fields for depth estimation from a single image. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 5162–5170, 2015.
- [75] Xiaopei Liu, Liang Wan, Yingge Qu, Tien-Tsin Wong, Stephen Lin, Chi-Sing Leung, and Pheng-Ann Heng. Intrinsic colorization. In ACM SIGGRAPH Asia 2008 Papers, SIGGRAPH Asia '08, 2008.
- [76] Yunfei Liu, Yu Li, Shaodi You, and Feng Lu. Unsupervised learning for intrinsic image decomposition from a single image. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3248–3257, 2020.



## Bibliography

---

- [77] Z. Lou, T. Gevers, N. Hu, and M.P. Lucassen. Color constancy by deep learning. In BMVC, pages 76.1–76.12, September 2015.
- [78] Ke Ma, Zhixin Shu, Xue Bai, Jue Wang, and Dimitris Samaras. Docunet: document image unwarping via a stacked u-net. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4700–4709, 2018.
- [79] Wei-Chiu Ma, Hang Chu, Bolei Zhou, Raquel Urtasun, and Antonio Torralba. Single image intrinsic decomposition without a single intrinsic image. In ECCV, 2018.
- [80] D. Mandl, K.M. Yi, P. Mohr, P.M. Roth, P. Fua, V. Lepetit, D. Schmalstieg, and D. Kalkofen. Learning lightprobes for mixed reality illumination. ISMAR, pages 82–89, 2017.
- [81] Matlab optimization toolbox, 2021. The MathWorks, Natick, MA, USA.
- [82] Yasuyuki Matsushita, Stephen Lin, Sing Bing Kang, and Heung-Yeung Shum. Estimating intrinsic images from image sequences with biased illumination. In Computer Vision - ECCV 2004, pages 274–286. Springer Berlin Heidelberg, 2004.
- [83] Yasuyuki Matsushita, Ko Nishino, Katsushi Ikeuchi, and Masao Sakauchi. Illumination normalization with time-dependent intrinsic images for video surveillance. IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(10):1336–1347, 2004.
- [84] Hubert Michalak and Krzysztof Okarma. Robust combined binarization method of non-uniformly illuminated document images for alphanumerical character recognition. Sensors, 20(10):2914, 2020.
- [85] Francesc Moreno-Noguer, Shree K Nayar, and Peter N Belhumeur. Optimal illumination for image and video relighting. In SIGGRAPH Sketches, page 75. Citeseer, 2005.
- [86] L. Murmann, M. Gharbi, M. Aittala, and F. Durand. A dataset of multi-illumination images in the wild. In ICCV, pages 4080–4089, 2019.
- [87] Takuya Narihira, Michael Maire, and Stella X. Yu. Direct intrinsics: Learning albedo-shading decomposition by convolutional regression. In International Conference on Computer Vision (ICCV), 2015.

- [88] Mamata Nayak and Ajit Kumar Nayak. Odia characters recognition by training tesseract ocr engine. International Journal of Computer Applications, 975:8887, 2014.
- [89] Thomas Nestmeyer and Peter V. Gehler. Reflectance adaptive filtering improves intrinsic image estimation. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1771–1780, 2017.
- [90] Thomas Nestmeyer, Jean-François Lalonde, Iain Matthews, and Andreas Lehrmann. Learning physics-guided face relighting under directional light. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5124–5133, 2020.
- [91] Addy Ngan, Frédo Durand, and Wojciech Matusik. Experimental analysis of brdf models. Rendering Techniques, 2005(16th):2, 2005.
- [92] A. Panagopoulos, D. Samaras, and N. Paragios. Robust shadow and illumination estimation using a mixture model. In CVPR, pages 651–658. IEEE, 2009.
- [93] A. Panagopoulos, C. Wang, D. Samaras, and N. Paragios. Illumination estimation and cast shadow detection through a higher-order graphical model. CVPR 2011, pages 673–680, 2011.
- [94] Matt Pharr, Wenzel Jakob, and Greg Humphreys. Physically based rendering: From theory to implementation. Morgan Kaufmann, 2016.
- [95] Julien Philip, Michaël Gharbi, Tinghui Zhou, Alexei A Efros, and George Drettakis. Multi-view relighting using a geometry-aware network. ACM Transactions on Graphics (TOG), 38(4):1–14, 2019.
- [96] K.N. Plataniotis and A.N. Venetsanopoulos. Color Image Processing and Applications. Springer, 2000.
- [97] Densen Puthussery, Melvin Kuriakose, Jiji C V, et al. Wdrn: A wavelet decomposed relightnet for image relighting. arXiv preprint arXiv:2009.06678, 2020.
- [98] Yvain Quéau, Bastien Durix, Tao Wu, Daniel Cremers, François Lauze, and Jean-Denis Durou. Led-based photometric stereo: Modeling, calibration and numerical solution. Journal of Mathematical Imaging and Vision, 60(3):313–340, 2018.

## Bibliography

---

- [99] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.
- [100] D. Samaras and D. Metaxas. Incorporating illumination constraints in deformable models for shape from shading and light direction estimation. PAMI, 25(2):247–264, 2003.
- [101] D. Schnieders. Light source estimation from spherical reflections. Hong Kong University, 2011.
- [102] Soumyadip Sengupta, Jinwei Gu, Kihwan Kim, Guilin Liu, David W Jacobs, and Jan Kautz. Neural inverse rendering of an indoor scene from a single image. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 8598–8607, 2019.
- [103] Marc Serra. Modeling, estimation and evaluation of intrinsic images considering color information. PhD thesis, Universitat Autònoma de Barcelona - Computer Vision Center, September 2015.
- [104] Marc Serra, Olivier Penacchio, Robert Benavente, and Maria Vanrell. Names and shades of color for intrinsic image estimation. In IEEE Conference on Computer Vision and Pattern Recognition, pages 278–285, 2012.
- [105] Steven A Shafer. Describing light mixtures through linear algebra. JOSA, 72(2):299–300, 1982.
- [106] Steven A Shafer. Using color to separate reflection components. Color Research & Application, 10(4):210–218, 1985.
- [107] Evan Shelhamer, Jonathan T. Barron, and Trevor Darrell. Scene intrinsics and depth from a single image. In The IEEE International Conference on Computer Vision (ICCV) Workshops, December 2015.
- [108] Li Shen and Chuohao Yeo. Intrinsic images decomposition using a local and global sparse representation of reflectance. In CVPR, pages 697 – 704, 07 2011.
- [109] Li Shen, Chuohao Yeo, and Binh-Son Hua. Intrinsic image decomposition using a sparse representation of reflectance. IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(12):2904–2915, 2013.

- [110] Jian Shi, Yue Dong, Hao Su, and Stella X. Yu. Learning non-lambertian object intrinsics across shapenet categories. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 5844–5853, 2017.
- [111] H. Sial, S. Sancho-Asensio, R. Baldrich, R. Benavente, and M. Vanrell. Color-based data augmentation for reflectance estimation. In IS&T Color and Imaging Conference, volume 2018, pages 284–289, 11 2018.
- [112] H.A. Sial, R. Baldrich, and M. Vanrell. Deep intrinsic decomposition trained on surreal scenes yet with realistic light effects. JOSA A, 37(1):1–15, 2020.
- [113] Hassan A. Sial, Ramon Baldrich, and Maria Vanrell. Deep intrinsic decomposition trained on surreal scenes yet with realistic light effects. J. Opt. Soc. Am. A, 37(1):1–15, Jan 2020.
- [114] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. Proceedings of 30th IEEE Conference on Computer Vision and Pattern Recognition, 2017.
- [115] Tiancheng Sun, Jonathan T Barron, Yun-Ta Tsai, Zexiang Xu, Xueming Yu, Graham Fyffe, Christoph Rhemann, Jay Busch, Paul E Debevec, and Ravi Ramamoorthi. Single image portrait relighting. ACM Trans. Graph., 38(4):79–1, 2019.
- [116] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2818–2826, 2016.
- [117] Marshall F Tappen, Edward H. Adelson, and William T. Freeman. Estimating intrinsic component images using non-linear regression. In IEEE Conference on Computer Vision and Pattern Recognition, pages 1992–1999, 2006.
- [118] Marshall F Tappen, William T. Freeman, and Edward H. Adelson. Recovering intrinsic images from a single image. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(9):1459–1472, 2005.
- [119] Shoji Tominaga. Using reflectance models for surface estimation. In Color and Imaging Conference, volume 1995, pages 29–33. Society for Imaging Science and Technology, 1995.

## Bibliography

---

- [120] Tomás F. Yago Vicente, Le Hou, Chen-Ping Yu, Minh Hoai, and Dimitris Samaras. Large-scale training of shadow detectors with noisily-annotated shadow examples. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, Computer Vision – ECCV 2016, 2016.
- [121] Bingshu Wang and CL Philip Chen. An effective background estimation method for shadows removal of document images. In 2019 IEEE International Conference on Image Processing (ICIP), pages 3611–3615. IEEE, 2019.
- [122] Li-Wen Wang, Wan-Chi Siu, Zhi-Song Liu, Chu-Tak Li, and Daniel PK Lun. Deep relighting networks for image light source manipulation. arXiv preprint arXiv:2008.08298, 2020.
- [123] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing, 13(4):600–612, 2004.
- [124] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003, volume 2, pages 1398–1402. Ieee, 2003.
- [125] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 8052–8060, 2018.
- [126] Yair Weiss. Deriving intrinsic images from image sequences. In International Conference on Computer Vision, pages 68–75, 2001.
- [127] Changchang Wu et al. Visualsfm: A visual structure from motion system. 2011.
- [128] Gunther Wyszecki and W. Stiles. Color science: Concepts and methods, quantitative data and formulae, 2nd edition. Color Research & Application, 07 2000.
- [129] J Y. Bouguet. Matlab camera calibration toolbox. IEEE Transactions on Reliability - TR, 01 2005.
- [130] S. Yan, F. Peng, H. Tan, S. Lai, and M. Zhang. Multiple illumination estimation with end-to-end network. ICIVC, pages 642–647, 2018.

- [131] Shaodi You, Yasuyuki Matsushita, Sudipta Sinha, Yusuke Bou, and Katsushi Ikeuchi. Multiview rectification of folded documents. IEEE transactions on pattern analysis and machine intelligence, 40(2):505–511, 2017.
- [132] Ye Yu and William AP Smith. Inverserendernet: Learning single image inverse rendering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3155–3164, 2019.
- [133] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 586–595, 2018.
- [134] Qi Zhao, Ping Tan, Qiang Dai, Li Shen, Enhua Wu, and Stephen Lin. A closed-form solution to retinex with nonlocal texture constraints. IEEE Trans. Pattern Anal. Mach. Intell., 34(7):1437–1444, July 2012.
- [135] Hao Zhou, Sunil Hadap, Kalyan Sunkavalli, and David W Jacobs. Deep single-image portrait relighting. In Proceedings of the IEEE International Conference on Computer Vision, pages 7194–7202, 2019.
- [136] Tinghui Zhou, Philipp Krähenbühl, and Alexei A. Efros. Learning data-driven reflectance priors for intrinsic image decomposition. 2015 IEEE International Conference on Computer Vision (ICCV), pages 3469–3477, 2015.