Universidad Zaragoza
1542

Trabajo Fin de Máster

Practical Intrinsic Image Decomposition

Autora

Elena Garcés García

Director

Diego Gutiérrez Pérez

Escuela de Ingeniería y Arquitectura

Septiembre 2012

Si oyes una voz dentro de ti diciéndote 'no sabes pintar', pinta, faltaría más, y la voz se callará. Vincent Van Gogh

Resumen

El conocimiento previo de las luces y los materiales que componen una escena es el primer paso para su total captura y reconstrucción. Sin embargo, obtener esta información a partir de una sencilla fotografía no es una tarea fácil. Cuando capturamos una imagen del mundo real, toda la información de color, geometría e iluminación se integra en el sensor de nuestra cámara dando como resultado un conjunto de píxeles RGB. Estos valores carecen de toda la información geométrica de la imagen que nos permitiría realizar tareas como reiluminación o cambio de materiales.

El objetivo de la presente Tesis Fin de Máster ha sido estudiar y resolver este problema que comúnmente se conoce como descomposición de una imagen en sus componentes intrínsecas, y que consiste en obtener, para una única imagen, la parte correspondiente a iluminación y la que corresponde con reflectancia (textura, color).

Actualmente, la mayoría de los métodos que resuelven este problema requieren excesiva interacción del usuario. De este modo, un usuario inexperto o la ausencia de información pueden dar lugar a malas descomposiciones. En este trabajo se ha tratado de obtener una solución eficiente, con resultados de alta calidad y robustos, partiendo de una única imagen de la escena a descomponer. En particular, se han estudiado dos soluciones distintas.

La primera solución propuesta, denominada *Intrinsic Images by Clustering*, ha sido publicada en la revista *Computer Graphics Forum* cuyo JCR 2011 index es 35/83 (Q2) en la categoría de *Computer Science, Software Engineering*, con un índice de impacto de 5 años de 1.634. El método propuesto requiere una única imagen para funcionar y se basa en la detección en la imagen de zonas de la misma reflectancia. Con esta información se construye un sistema de ecuaciones lineales donde se describen las conexiones y las relaciones entre ellas. Este algoritmo constituye el actual estado del arte en métodos de separación en imágenes intrínsecas a partir de una sola imagen de entrada.

La segunda solución planteada ha sido desarrollada en colaboración con la empresa Adobe Systems Inc. bajo la supervisión del Dr. Sunil Hadap. El nuevo método se basa en la observación de que los gradientes de reflectancia de la imagen siguen una dirección invariante y relativa a la fuente de luz. De este modo, estimando la dirección invariante a partir de la información de color de la imagen, podríamos ser capaces de desambiguar los cambios debidos a reflectancia y los cambios debidos sombreado. Los resultados obtenidos de este primer estudio del algoritmo bajo un entorno controlado concluyen que el algoritmo tiene mucho potencial, y se abre una interesante vía para futuras investigaciones mediante la combinación con otras técnicas complementarias que aporten nueva información de la escena.

Descomponer una imagen en sus componentes intrínsecas es todavía un problema abierto con múltiples aplicaciones potenciales. Con esta investigación se ha contribuido con un paso más hacia la solución global y óptima. Además, se concluye que futuras investigaciones deberían enfocarse a obtener un algoritmo que requiera la menor interacción posible, ya que debido a la complejidad del problema es matemáticamente imposible obtener una solución única y sin interacción para todos los escenarios.

Contents

Prologue

The problem we study in this thesis was initially explored in the PFC titled *Descomposición de imágenes en sus componentes intrínsecas*. However, the underlying assumptions and the proposed algorithm have changed considerably. Most relevant results of the PFC were accepted for publication in V Ibero-American Symposium in Computer Graphics 2011. In that paper [GGLM11] and PFC, we explored several clustering techniques which were not used in the following research.

The first method exposed in this master thesis, *Intrinsic Images by Clustering*, was presented at Eurographics Symposium on Rendering 2012 in Paris, France. A total of 21 out of 69 papers were accepted for the conference, the acceptance rate being 30%. Articles accepted for this conference are also published in the journal Computer Graphics Forum, whose JCR 2011 index is 35/83 (Q2) in the category of Computer Science, Software Engineering (5-year impact factor 1.634). The first part of this master thesis contains essentially the same information as the paper, extending the explanation of the problem and its background, and adding certain data and information which, due to limited space, did not appear in the article. The article can be found in the Appendices of this master thesis.

The second method, *Intrinsic Images by Co-linear Albedo*, resulted from a three months internship at San Jose, California (EEUU) and a six months collaboration with the company Adobe Systems Corp. under the supervision of Dr. Sunil Hadap. The description of the algorithm and results are presented in the second part of this master thesis. During that collaboration, I also worked with Jorge Lopez-Moreno, Erik Reinhard and Diego Gutierrez on another paper currently submitted to the aforementioned journal *Computer Graphics Forum*.

Chapter 1

Introduction

1.1 Motivation

The field of computational photography has emerged from the necessity of algorithms and techniques to improve common image processing operations, such as, image relighting, material editing or 3D reconstruction. There has been a huge research effort in this field trying to obtain intrinsic information from the scene using, for example, coded apertures, depth cameras or stereo vision. However, the applicability of these techniques is limited by the available input information, which depends directly on the technology used to capture the images. Unfortunately, the use of this technologies is not generalized and the most common scenario is to have a single image taken by a conventional digital camera.

When we take a picture of the real world, all the information about about geometry, lights, material and color is integrated into a single RGB pixel. Thus, changing inherent properties of a scene such as lighting or materials, is a hard task that must be done by hand, pixel per pixel, by artists and experts user on tools like *Photoshop*.

The problem of obtaining lighting and materials from a single image has been an open challenge since it was first formulated with the name of intrinsic image decomposition [BT78]. This decomposition consists of separating an input image in two components: one image representing the reflectance or albedo of the object and other image representing the shading or illumination. We can see an example of this decomposition in Figure 1.1.

This problem is usually formulated as the input image I being a per-pixel product of its unknown intrinsic shading S and reflectance R, so the space of mathematically valid solutions is in fact infinite. Given a single image, the problem is extremely ill-posed e.g. a dark-blue pixel can come from, a white material illuminated by a dark blue light, or from a dark-blue material illuminated by a bright white light. Existing methods therefore need to rely on additional sources of information, such as making reasonable assumptions about the characteristics of the intrinsic components, having multiple images under different illuminations or asking the user to add image-specific input.

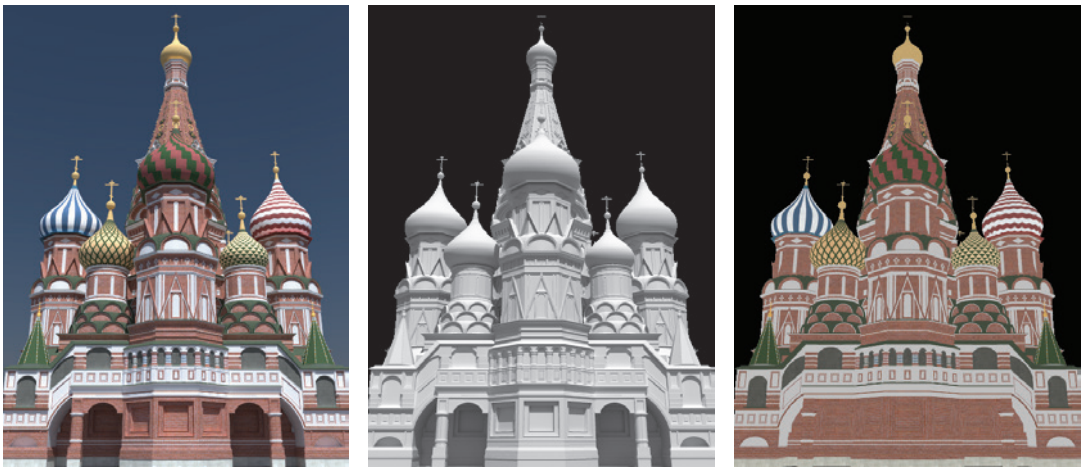


Figure 1.1: *Ground truth intrinsic images, shading and reflectance. Notice that the shading image is a monochromatic image representing the illumination and shadows while the reflectance image is a plain colored texture without any information about shadows or geometry.* Images provided by Pierre-Yves Laffont.

1.2 Approach and contributions of this work

This research has focused on the problem of obtaining the intrinsic images from a single image without any user interaction. We have devised two different solutions. The first solution, *Intrinsic Images by Clustering*, is a segmentation-based algorithm which has been proved to work in wide range of scenarios. Although the proposed method is currently the state of the art in automatic methods, there is still room for improvement. For that reason, we explored a second solution, *Intrinsic Images by Co-linear Albedo*, which consists on an per pixel-based optimization.

Intrinsic Images by Clustering (Chapter 3)

In this approach, we do make some reasonable assumptions, in the form of flexible constraints. We formulate the decomposition of an input image into its shading and reflectance components as a linear system that exploits relations between clusters of similar reflectance. Classic Retinex approaches assume that i) reflectance is piecewise constant, and ii) shading is spatially smooth (C^0 and C^1 continuity) [LM71, Hor74]. Based on this, a number of authors have proposed different approaches [FDB92, KES$^+$03, STL08, GRK$^+$11]. In this work we first find clusters of similar reflectance in the image following the observation that *changes* in chromaticity usually correspond to changes in reflectance.

We then relax the second Retinex assumption that shading is spatially smooth in two ways: we assume only C^0 continuity on the shading, and only *at the boundaries* between clusters (as opposed to the whole image), and describe this as a set of linear equations. Our linear system is completed by additionally preserving reflectance between clusters even if they are not contiguous, and adding a regularization term to make it more stable.

Our main contribution is a novel algorithm for intrinsic images decomposition which deals with a wider range of scenarios than traditional Retinex-based algorithms, yields better decompositions than existing automatic methods from single images, and offers an attractive trade-off between quality and ease of use, compared with techniques requiring either significant user input or multiple input images. We present an exhaustive comparison against most existing techniques (see Appendix A), which we will make public along to our source code (once the paper is published). Last, we show compelling example applications of retexturing, relighting and material editing based on our results.

Like all existing methods that deal with this ill-posed problem, our work is not free of limitations: Our C^0 assumption is a simplification that breaks for some occlusion boundaries and sharp edges, which translate into inaccurate equations in the system. However, given our robust formulation which usually translates into a few thousand equations, these inaccurate equations represent a very small percentage, and our method generally handles these situations well.

Intrinsic Images by Co-linear Albedo (Chapter 4)

This second method was developed in collaboration with Dr. Sunil Hadap from Adobe Systems Inc. We decided to constrain the problem by making some assumptions about the reflectance and shading, following a similar path to previous works that assume that shading is a smooth function (C^0 and C^1 continuity) [LM71, Hor74]. Although this assumption limits the range of application to globally convex objects and flat surfaces, this is enough for our first study of the technique.

The solution relies on the observation that reflectance gradients of the image are almost parallel and follow an invariant direction. Thus, approximating this direction with the RGB pixel values of the image, we can differentiate changes in reflectance from changes in shading to obtain the solution. The problem is that this is not always possible and additional constraints are required. We then leverage the use of the Laplacian operator to guarantee a smooth shading function at every point.

We demonstrate that the solution works well for the proposed scenario but additional cues are necessary for more complex scenes. Thus, this conclusion opens the door to an interesting direction for future research.

Chapter 2

Related Work

Throughout the years the problem of intrinsic image decomposition has been tackled from different perspectives. In this chapter, we present a summary of the most important ones classified in three groups: *Automatic* methods, which work from one single image as input and may require few parameters to be tunned by the user; *User Assisted* methods, that require the user to contribute, normally with painted strokes, to generate the solution; and *Multiple Images*, in this group we include methods that rely on other information, either from multiple images or from alternative sources.

2.1 Automatic

Some automatic methods rely on reasonable assumptions about the nature of these two terms, or the correlation between different characteristics of the image. Horn [Hor74] presents a method to obtain lightness from black and white images, using pixel intensity information and assuming that lightness corresponds to reflectance. He further assumes that the reflectance remains locally constant while illumination varies smoothly (as described by the Retinex theory [LM71]). Funt et al. [FDB92] extend this approach to color images, and propose the analysis of chromaticity variations in order to identify the boundaries of different reflectance areas. They enforce integrability of the shading at these boundaries and propagate their values to their neighboring pixels by diffusion, solving the subsequent Poisson equation with a Fourier transformation. This was later extended by Shen et al. [STL08] with global texture constraints, forcing distant pixels with the same texture to have the same reflectance. This constraint greatly improves the performance of the standard Retinex method, although it relies on objects with repeated texture patterns and may yield posterization artifacts due to the wrong clustering of distant pixels.

The related method by Finlayson and colleagues [GDFL04] is mainly oriented to remove shadows by minimizing entropy, but does not to recover intrinsic images. The work by Jiang et al. [JSW10] assumes that correlated changes in mean luminance and luminance amplitude indicate illumination changes. By introducing a novel feature, *local luminance amplitude*, the authors obtain good results, although limited to images of relatively flat surfaces and objects from the recently published MIT dataset for intrinsic image decomposition [GJAF09]. This actually simplifies the problem since such objects are treated in

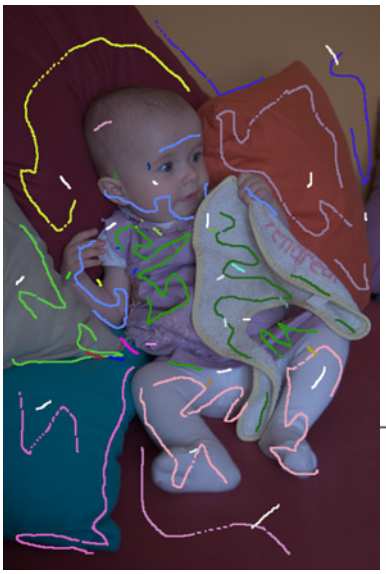
isolation, avoiding the problem of occlusion boundaries at the outlines. Recently, Gehler and colleagues[GRK+11] proposed a probabilistic model, based on a gradient consistency term and a reflectance prior, which assumes that reflectance values belong to a sparse set of basis colors. The problem is formulated as an optimization of the proposed energy function. The method yields good results, although again limited to isolated objects from the MIT dataset.

2.2 User intervention

Another set of techniques rely on assumptions *and* user intervention. Bousseau and colleagues [BPD09] simplify the problem by assuming that local reflectance variations lie in a 2D plane in RGB space not containing black, which may not be compatible with certain texture or grayscale images. This assumption is also used in the work by Shen and Yeo [SY11], who further consider that neighboring pixels in a local window with similar intensity have also similar reflectance. In addition, Bousseau's method requires that the user define constraints over the image by means of three different types of strokes: constant-reflectance, constant-illumination and fixed-illumination. Their method produces very compelling results, although creating the appropriate strokes for each particular image (from 15 to 81 strokes for the images in the document, see Figure 2.1) may be far from intuitive for unskilled users. The same set of user tools is employed in the recent work by Shen and colleagues [SYJL11], who use an optimization approach that further assumes that neighboring pixels with similar intensity have similar reflectance values. In the context of material editing, Dong et al. [DTPG11] assume input images of globally flat surfaces with small normal perturbations and lit with a directional light, and require user strokes for optimal decompositions.

2.3 Multiple images

Last, another strategy consists of incorporating additional information, either from other sources or from multiple images. Tappen et al. [TFA05] classify the derivatives of the image as produced either by illumination or albedo. Ten classifiers are obtained from training Adaboost [FS95] with a set of computer generated images containing only reflectance or illumination components. They further refine their approach by introducing a new training set of real-world images and including a method to weigh the response to these classifiers [TAF06]. Despite these advanced techniques, several configurations of illumination and reflectance remain very difficult to decompose and additional techniques like Markov Random Fields (MRF) and Belief Propagation (BP) are necessary in order to yield good solutions. Weiss [Wei01] uses a large sequence of images of the same scene (up to 64 images, and no less than 35, taken in controlled settings), where the reflectance remains constant and illumination varies in time. Also using multiple images, Laffont and colleagues [LBD11] leverage multi view stereo techniques to approximately reconstruct a point cloud representation of the scene. After some user intervention, illumination information computed on that point cloud is propagated in the image. Their method decomposes the illumination layer into three components: sun, sky and indirect



(a)

(b)

(c)

(d)

	Res.	User strokes	Matrix (s)	Solving (s)
Baby (fig. 1)	533 × 800	58	20.64	11.93
Flower (fig. 4)	750 × 500	15	18.21	7.87
Clown (fig. 10)	486 × 800	33	18.83	9.53
St Basile (fig. 14)	800 × 600	81	23.29	14.56
Paper (fig. 14)	750 × 500	36	18.17	9.2

(e)

Figure 2.1: *User Interaction of Bousseau et al. [BPD09]. (a) User strokes needed to obtain the decomposition of the input image (a) into the intrinsic images: (c) reflectance and (d) shading. (e) Number of strokes required for some of the images shown in the document. They use three different types of strokes: one to select regions with constant shading, one for constant reflectance, and a last one to identify regions where both shading and reflectance are constant.*

light. Last, the concept of *intrinsic colorization* is introduced by Liu et al. [LWQ+08]; to colorize a grayscale image, their method recovers the needed reflectance component from multiple images obtained from the web, in order to transfer color from there.

Chapter 3

Intrinsic Image Decomposition by Clustering

The desired decomposition consists of separating an image into two components (images): one representing reflectance information, and another containing the illumination or shading. We use RAW or linearized RGB values as input. For a Lambertian scene, the problem can be simply formulated as:

$$I(x, y) = S(x, y) * R(x, y) \tag{3.1}$$

where $I(x, y)$ is the input image, $S(x, y)$ is the shading image, $R(x, y)$ represents reflectance information and $*$ is a per-channel Hadamard product. Our goal is to obtain $S(x, y)$ and $R(x, y)$, given $I(x, y)$. We make the problem tractable with a few assumptions well-grounded on existing vision and image processing techniques. While of course our assumptions may not always be accurate throughout the whole image, they allow us to devise a method that works very well on a large range of images while keeping our algorithm simpler than other approaches.

3.1 Assumptions

Horn made the key observation that, for grayscale images, sharp reflectance changes cause intensity discontinuities in the luminance of an image [Hor74]. Our first assumption relies on the later generalization to color images by Funt et al. [FDB92], who associate changes in reflectance with changes in chromaticity. We first leverage this correlation between reflectance and chromaticity values by detecting regions of similar chromaticity in the input image, which are assumed to approximate regions of similar reflectance. We implement this as a soft constraint, though, which we relax in specific cases (see Sections 3.3 and 3.4).

Furthermore, existing Retinex-based techniques (see for instance [FDB92, KES+03, STL08]) assume that shading is a smooth function, therefore being both C^0 and C^1 continuous. However, there are a number of particular cases (such as some occlusion boundaries) in which this assumption does not hold. Our second assumption relaxes this restriction by imposing only C^0 continuity *at boundaries* between the regions previously

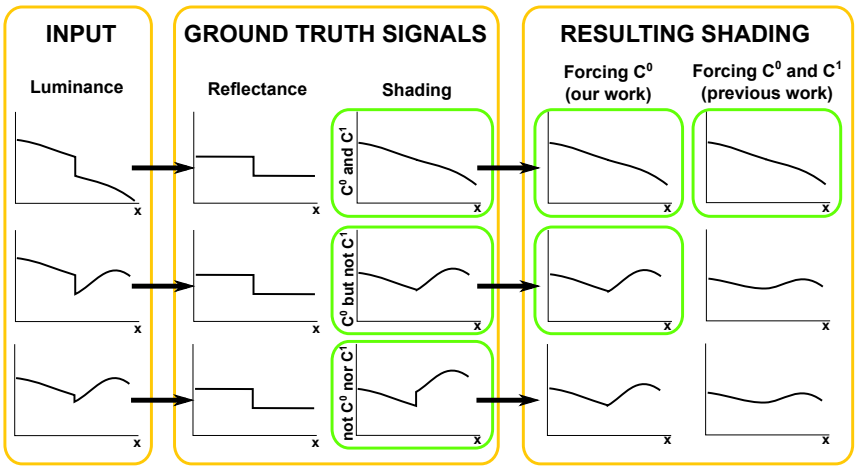


Figure 3.1: *Intrinsic shading estimation when both the shading and the reflectance present a discontinuity at the same point (as in some occlusion boundaries). Left column: Three different input luminance signals. Middle columns: Ground truth intrinsic signals. All three input signals are the result of multiplying the same reflectance with three different shading signals, presenting different continuity characteristics. Right columns: Results assuming both C^0 and C^1 continuity on the shading, compared to C^0 only (our method). Notice how our algorithm leads to an accurate result in two of the three cases, while yielding less error in the most unfavorable case.*

detected. This allows us to handle a wider variety of cases correctly; in cases where the smooth shading assumption does hold, our method naturally maintains C^1 continuity as well (see Figure 3.1). In cases where this assumption breaks, our method still provides a more accurate reconstruction of the intrinsic signals. Last, as previous works, we assume a white light source and a correctly white balanced input image.

3.2 Overview

Figure 3.2 shows an overview of our algorithm, applied to patches of different colors with a continuous shading gradient. It works in two steps, which we term *clustering* and *system definition*. First, we segment the input image according to chromaticity. We then subdivide the resulting segments, and obtain a set of clusters of connected pixels with similar chromaticity. This clustering is then refined to better approximate reflectance (as opposed to chromaticity) discontinuities, according to our first assumption.

Based on this clustering, we then build a linear system of equations defining the connections and relations between the different clusters, as well as the different constraints. One set of equations describes the C^0 continuity in the shading at cluster boundaries (our second assumption). We then make the observation that all clusters originally coming from the same segment should in principle maintain similar reflectance, even if they are not contiguous. This is similar to the observation made by Shen et al. [STL08]; however, we improve this in two important ways: first, we do not need to rely on texture information; second, we work at cluster level, as opposed to pixel level, which translates into a more stable solution. This yields our second set of equations. The system is completed with an additional regularization term. By solving the resulting linear system we obtain

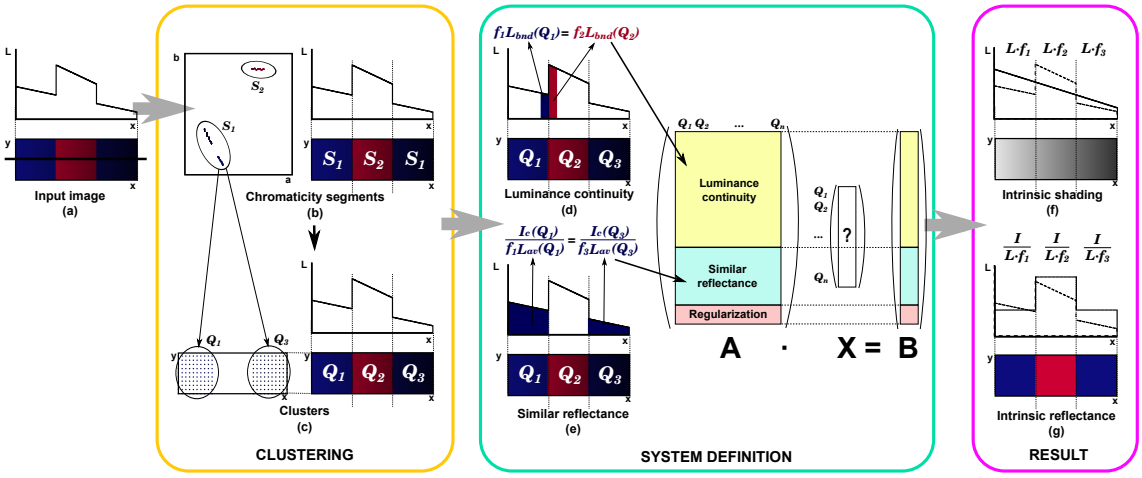


Figure 3.2: *Overview of the algorithm for the simple case of three colored patches and a continuos shading gradient.* ***(a)*** *Input image, with a plot of the pixels luminance along a scan line.* ***(b)*** *Initial k-means segmentation. Left: A scatter plot of the (a, b) coordinates (Lab color space) shows two segments of different chrominance (S_1 and S_2). Right: These segments belong to different parts of the image, with S_1 split in two image areas (labeled accordingly in the figure).* ***(c)*** *Subsequent clustering. Left: Segments are further divided into clusters of contiguous pixels. The example shows S_1 being clustered into Q_1 and Q_3 in image space. Right: clusters labeled in the image.* ***(d)*** *Enforcing luminance continuity on the boundaries between two clusters yields a large number of equations for the linear system.* ***(e)*** *Clusters originally belonging to the same segment maintain similar reflectance properties (the example shows Q_1 and Q_3, both belonging initially to S_1). This yields another set of equations. The final system is completed with a regularization term.* ***(f)*** *Result: Intrinsic shading. It is a continuous signal, as described by the equations in (d).* ***(g)*** *Result: Intrinsic reflectance. Q_1 and Q_3 share the same reflectance, as described by the equations in (e). Please refer to the text for further details on the equations and their notation.*

the intrinsic shading image; reflectance is obtained by means of a simple per-pixel RGB division (Equation 3.1), as previous works [BPD09]. The next sections describe these steps in detail.

3.3 Clustering

We aim to divide the image into clusters of similar chrominance properties. Given our assumptions, the boundaries between those clusters will indicate reflectance discontinuities. This is a reasonable task, given the reduced set of reflectance values in natural images [OW04]. This reduced set was also leveraged in recent work by Bousseau et al. [BPD09], who further assumed that reflectance colors lie in a 2D plane not intersecting the origin. Several existing segmentation techniques, such as Mean Shift, the graph-based method by Felzenszwalb and Huttenlocher [FH04] or its subsequent modification [GGLM11] have been thoroughly tested, but unfortunately none would yield satisfying results for our purposes. We thus have designed a novel two-step clustering strategy, specially tailored for the problem of intrinsic images decomposition. For the sake of clarity, we refer to the first step as *segmentation*, and to the second as *clustering*.

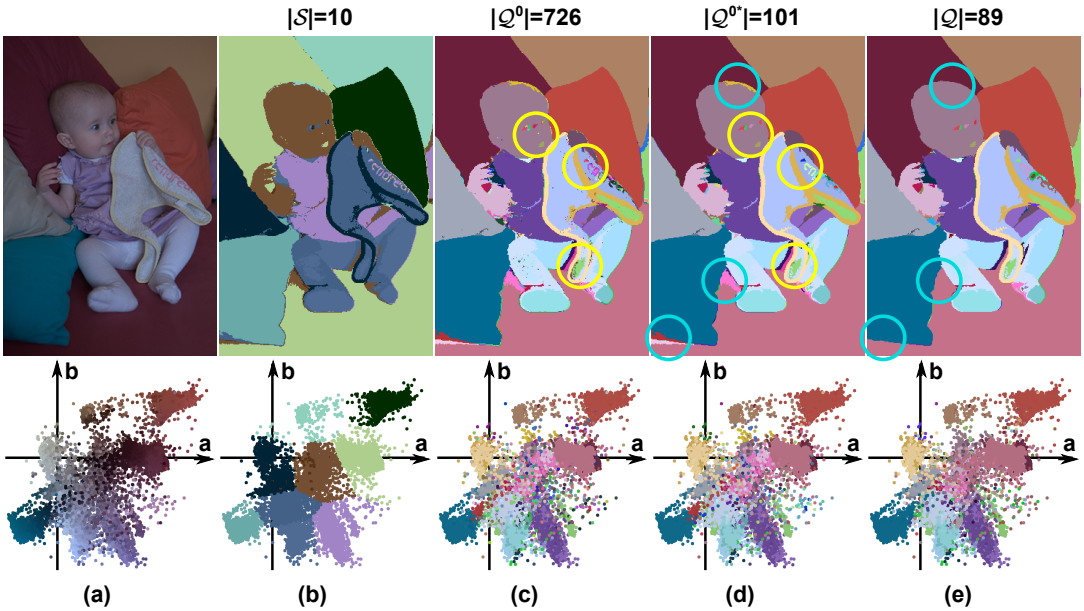


Figure 3.3: *Our segmentation-clustering process. For all the images the corresponding scatter plot in the (a,b) plane is represented at the bottom. (a) Input image (b) Result of the first segmentation step, yielding 10 distinct segments in $\mathcal{S}$. (c) First segmentation step. (d) Initial clustering (726 clusters in $\mathcal{Q}^o$). (e) Merging small clusters. (f) Final cluster set $\mathcal{Q}$ after merging smooth boundaries (89 clusters). The yellow and blue circles highlight areas were the effects of these last two steps are clearly visible. Notice how noise is eliminated (yellow circles), as well as smooth gradients due to shadows (blue circles).*

Segmentation

We first segment the image according to chromaticity values, regardless of the spatial location of the pixels. We define our segmentation feature space as $\mathcal{F} = \{\beta, a, b\}$ where (a, b) are the chromatic coordinates of the input image in $CIELab$ space, and β is a feature defined to handle strong blacks or whites (these are defined as pixels with very low chromaticity values and very low or high luminance). These values would be difficult to segment properly in a chromaticity-based algorithm, and usually describe important reflectance features. For each pixel in the image, we define β as:

$$\beta = \begin{cases} -\mu & \text{if } (|a| < \lambda) \;\&\; (|b| < \lambda) \;\&\; (L < L_{min}) \\ +\mu & \text{if } (|a| < \lambda) \;\&\; (|b| < \lambda) \;\&\; (L > L_{max}) \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

where $\mu = 10^5$, $\lambda = 0.20 \max(|a|, |b|)$, $L_{min} = 0.15 \max(L)$ and $L_{max} = 0.95 \max(L)$. For this initial segmentation, we use the k-means implementation from Kanungo et al. [KMN+04]. Gehler and colleagues [GRK+11] also used k-means for their global sparse reflectance prior, which along with their shading prior and their gradient consistency term, fit into their global optimization system. In contrast, we use this segmentation to drive a simple and efficient system of linear equations. Note that the high μ value in the definition of β in equation 3.2 effectively forces the algorithm to create different segments with only strong black (or white) pixels. Except otherwise noted, we set $k = 10$ as the number of segments, but in our implementation it is left as a user parameter. The result of this step is a set of segments $\mathcal{S} = \{S_i\}$ (see Figures 3.3.a and 3.3.b). These will guide the clustering step of

the process, and help define global reflectance constraints between disconnected areas of the image during the *system definition* stage of the algorithm (Section 3.4).

Clustering

The previous segmentation defines global relations between (possibly disconnected) regions of the image. We now take into account local constraints by considering spatial contiguity. We first subdivide each segment $S_i \in \mathcal{S}$ into a set of *clusters* of contiguous pixels (8-neighborhood in image space), obtaining $\mathcal{Q}^o = \{Q_i^o\}$ (Figure 3.3.c). This set $\mathcal{Q}^o$ may contain very small clusters (due to quantization, aliasing or smooth varying textures), which could potentially later make our system less stable, or pairs of connected clusters where changes in chromaticity do not correspond to changes in reflectance (maybe due to shadows [GDFL04]).

Merging small clusters: Given a cluster Q_r^o containing less than p pixels, we locate its neighbor cluster Q_s^o with the closest average chrominance and merge them together: $Q_s^{o*} = Q_r^o \cup Q_s^o$. For the results in this thesis we use $p = 10$. This process is iterated until no more small clusters remain (Figure 3.3.d).

Merging smooth boundaries: Since chrominance and reflectance are not always exactly related, the k-means algorithm might yield over-segmented results. Given two adjacent clusters Q_r^{o*} and Q_s^{o*}, we average RGB pixel differences across the common border, obtaining a scalar d. The clusters are merged into Q_{rs} if $d < D$. The threshold D is set to 0.01 times the maximum pixel value in the image.

The result after these operations is our final cluster set $\mathcal{Q} = \{Q_i\}$ (see Figure 3.3.e).

3.4 System definition

The previous step has yielded a set of clusters separated by reflectance discontinuities. We now describe how to estimate the intrinsic shading from this initial clustering. We define a per-cluster factor f_i that, multiplying the luminance of the pixels of the cluster, will result into the intrinsic shading:

$$S(x, y) = f_i L(x, y) \tag{3.3}$$

where $(x, y) \in Q_i$. Instead of using expensive optimization techniques, we build a linear system of equations where f_i are the unknowns of our system. This system is built from three sets of equations as described below.

Luminance continuity We first enforce C^0 luminance continuity at the boundaries between clusters, in effect assigning abrupt changes at such boundaries to reflectance variations. Given the boundary between two clusters Q_r and Q_s:

$$f_r \, L_{bnd}(Q_r) - f_s \, L_{bnd}(Q_s) = 0 \tag{3.4}$$

where $L_{bnd}(Q_r)$ represents the luminance of the pixels in cluster Q_r at the boundary with cluster Q_s (and vice versa for $L_{bnd}(Q_s)$, see Figure 3.2.d). Last, f_r and f_s are the unknowns that force luminance continuity. In practice, we make Equation 3.4 more robust and obtain $L_{bnd}(\cdot)$ by averaging the luminance values of several pixels in a small window to each side of the boundary. We set the width of this window to three pixels for all the images in the paper.

However, applying exactly Equation 3.4 leads to an unstable behavior of the linear system; instead, we rewrite it in log-space:

$$\ln(f_r) - \ln(f_s) = \ln\left(\frac{L_{bnd}(Q_s)}{L_{bnd}(Q_r)}\right) \tag{3.5}$$

which leads to a more stable system and avoids both the trivial solution $f_i = 0$ and solutions with any $f_i < 0$. We apply Equation 3.5 to each pair of contiguous clusters.

Clusters of similar reflectance All clusters in $\mathcal{Q}$ coming from the same segment $S_i \in \mathcal{S}$ should in principle maintain similar reflectance. For each pair of clusters $\{Q_r, Q_s\} \in S_i$ we then have one equation per-channel with $c = \{R, G, B\}$:

$$\frac{I_c(Q_r)}{f_r\, L_{av}(Q_r)} = \frac{I_c(Q_s)}{f_s\, L_{av}(Q_s)} \tag{3.6}$$

where $I_c(r)$ is pixel average of the input image for all the pixels of the cluster Q_r and $L_{av}(r)$ is the average luminance of cluster Q_r (with an analogous definition for Q_s). We again reformulate this in log-space:

$$\ln(f_s) - \ln(f_r) = \ln\left(\frac{I_c(Q_s)L_{av}(Q_r)}{I_c(Q_r)L_{av}(Q_s)}\right) \tag{3.7}$$

However, clusters of the same chromaticity may actually have different reflectance, in which case the corresponding equations should not be included in the system. We adopt a conservative approach, and turn to the L coordinate to distinguish between different reflectances (e.g. light red and dark red). We define a threshold T_L of 5% of the maximum luminance of the image, and apply Equation 3.7 across clusters only if $|L_{av}(Q_r) - L_{av}(Q_s)| < T_L$.

Luminance regularization Last, we add a regularization equation to the system as follows:

$$\sum_i \ln f_i = 0 \tag{3.8}$$

With these three steps, we have posed our problem of obtaining intrinsic images as a system $\mathbf{AX} = \mathbf{B}$, where the number of equations equals the number of cluster boundaries plus the reflectance similarity equations plus the regularization equation. The elements of the unknown vector $\mathbf{X}$ are $x_i = \ln(f_i)$. In order to ensure the numerical stability of the solution, we solve the equivalent system $\mathbf{A^T AX} = \mathbf{A^T B}$ by means of a Quasi-Minimal Residual method (QMR) [BBC+94]. We found that using a standard Jacobi pre-conditioner yields good results in our context. From the solution vector $\mathbf{X}$ we can trivially obtain the final $f_i = \exp^{x_i}$ for each cluster Q_i. Last, we account for the fact that

our algorithm assigns each pixel to a given cluster (thus creating hard borders between clusters), while in contrast all the input images present some blur at the edges due to the imaging process; we therefore apply a 5×5 median filter at the cluster boundaries to obtain the final shading image.

3.5 Results and Discussion

Most of the results shown in this thesis have been generated from 16-bit RAW images, although in general our algorithm works well on linearized, 8-bit images. Larger versions of the results, along with detailed clustering and scatter plot data, are included in the Appendix A. Using our unoptimized code, our algorithm works at interactive rates e.g. an average image such as *clown* (see Figure 3.8), which results in 834 clusters, takes around 5 seconds on an Intel Core i5-2500 CPU at 3.30 GHz. In our tests, the most complex images may have up to 3000 clusters, resulting in about 15 seconds of processing time. Figure 3.4 shows how our algorithm successfully deals with the varied geometrical and texture complexities of several challenging web images, producing satisfying results (more examples can be found in the Appendix A). *Dragon* in the last row illustrates how 8-bit images may present pixel values close to zero, which cause numerical instabilities and are hard to disambiguate for any intrinsic images decomposition algorithm. This problem is greatly ameliorated with 16-bit images.

In Figure 3.8 we include an exhaustive comparison against state-of-the-art methods. We have tried to gather together all the published results common to most methods, but not all methods report results on all images. Compared to other automatic, single-image approaches, Tappen et al.'s technique [TFA05] shows perceivable reflectance artifacts in the shading image. Shen et al.'s method [STL08] suffers from posterization artifacts in the recovered reflectance (see for instance the *doll* image or the sky in *St. Basil*); additionally, almost all the shading images show severe continuity artifacts.

Shen and Yeo [SY11] provide a somewhat limited selection of images in their paper, both of which appear in our figure. It can be seen how in *baby*, the shading shows inconsistencies such as the exaggerated contrast between the legs and floor, while the eyes have been wrongly assigned to the shading layer.

The method of Gehler et al. [GRK+11], although producing reasonable results for the MIT dataset, tends to retain reflectance in the shading layer for natural images (see for instance *baby* image in Figure 3.8 or *synthetic doll* in the Appendix A). Moreover, their optimization function is computationally expensive, taking several hours, while our technique takes seconds to finish.

Last, the *doll* result using Weiss's approach [Wei01], despite using 40 images taken under carefully controlled settings, shows clear shading residuals in the reflectance layer, and wrong gradients in the shading image.

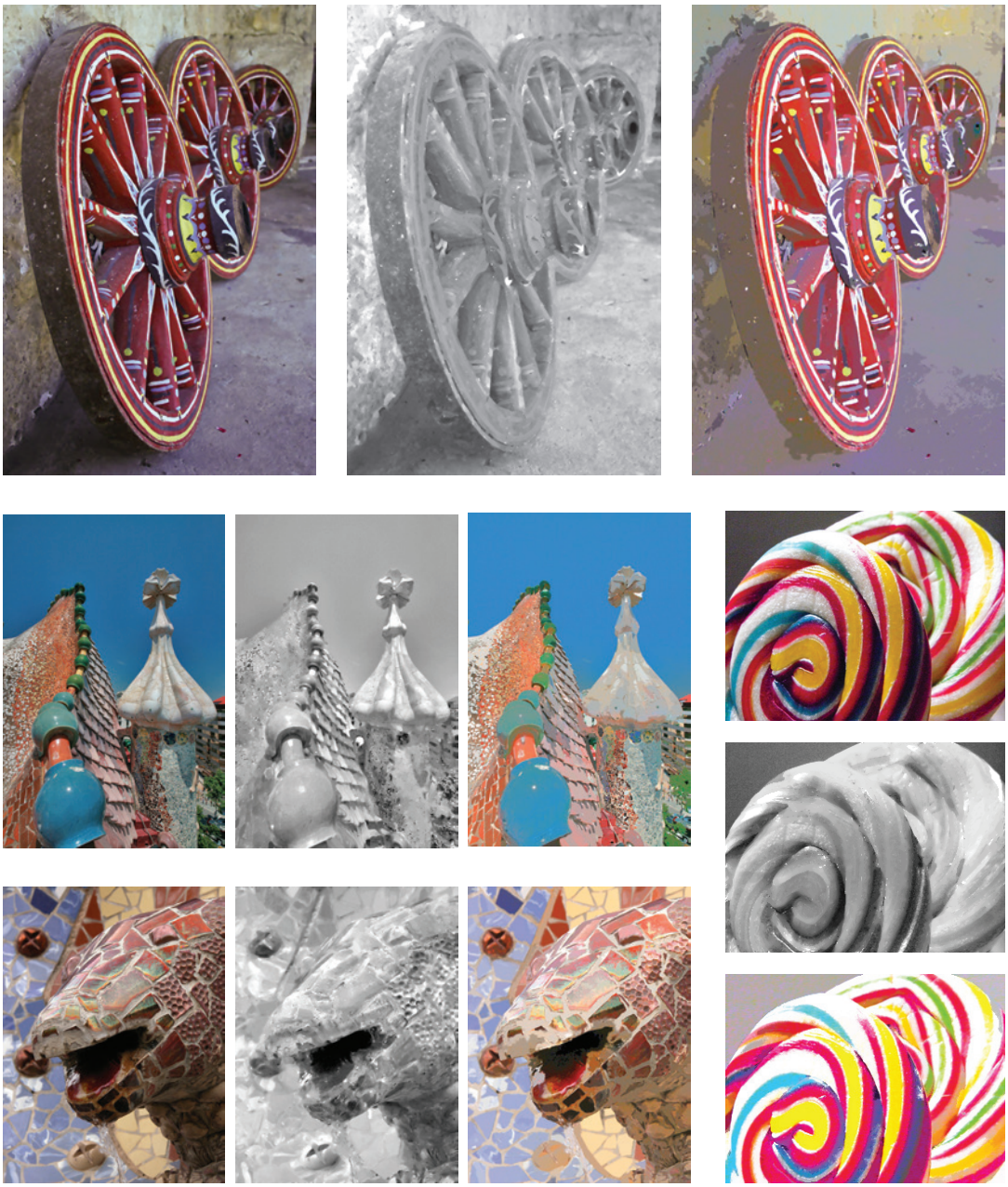


Figure 3.4: *Intrinsic images obtained with our method (using 8-bit input images). From left to right or from top to down: Input image, intrinsic shading and intrinsic reflectance. First row: wheels,* k = 16 *(original image by Angela Smith Kirkman). Second row: Batlló house,* k = 12 *(original image by lukasz dzierzanowski, flickr.com). Last row: dragon,* k = 22 *(original image by Jordanhill School D&T Dept, flickr.com). Left-down column: lollipop,* k = 10 *(original image by Thalita Carvalho, flickr.com)*

Our work yields results on-par with the user-assisted techniques by Bousseau et al. [BPD09] and Shen et al. [SYJL11], without requiring any user strokes. In *St. Basil*, our solution shows some artifacts especially visible in the black areas assigned to reflectance. Bousseau's result is free from such artifacts, although at the expense of requiring 81 user strokes, divided in three different kinds. Although the authors show that their method is robust

to small perturbations applied to such strokes, placing them from scratch is probably challenging for unskilled users. In contrast, our automatic method does a better job at assigning the doll's eyelashes to the reflectance layer Furthermore, Bousseau's method is not well fitted for texture images with rich reflectance variations, as recently demonstrated by Dong and colleagues [DTPG11]. Our approach is free from such restriction, and it handles those cases well (see Figure 3.6 and Figure 10 in the Appendix A).

Applications

Accurate decomposition into intrinsic images can play an important role in a broad range of applications such as material editing or image relighting. Figure 3.5 shows some applications using our intrinsic decomposition. For re-texturing we modify the reflectance layer and use the same normal-from-shading approach to deform the new textures. The relighting example, in this case, uses an image-based relighting algorithm [LMHRG10] that modifies the shading image before multiplying it by a sepia-shifted reflectance. Last, the wheel on the right shows another example of global reflectance manipulation. Furthermore, our technique can also be used in conjunction with others, such as the image-based material modeling technique recently presented by Dong et al. [DTPG11].

Figure 3.5: *Image edits accomplished using our intrinsic decompositions. Left: Re-texturing by editing the intrinsic reflectance. Center: Relighting of the previous result editing the intrinsic shading (sepia effect by editing intrinsic reflectance). Right: Global color edits on the intrinsic reflectance.*

MIT dataset

We have also tested our method using images from the MIT image dataset provided by Grosse et al. [GJAF09]. This dataset is designed to test intrinsic image decomposition methods, providing ground truth images for a variety of real-world objects. Figure 3.6 shows some examples of our results, compared against Color-Retinex, the combination of Weiss [Wei01] and Retinex and the recent approach by Dong et al. [DTPG11]. Note that Weiss's technique requires more than 30 input images, while the algorithm by Dong et al. requires user strokes. As we can observe, our algorithm obtains near optimal results in these cases. Please refer to the Appendix A for more results.

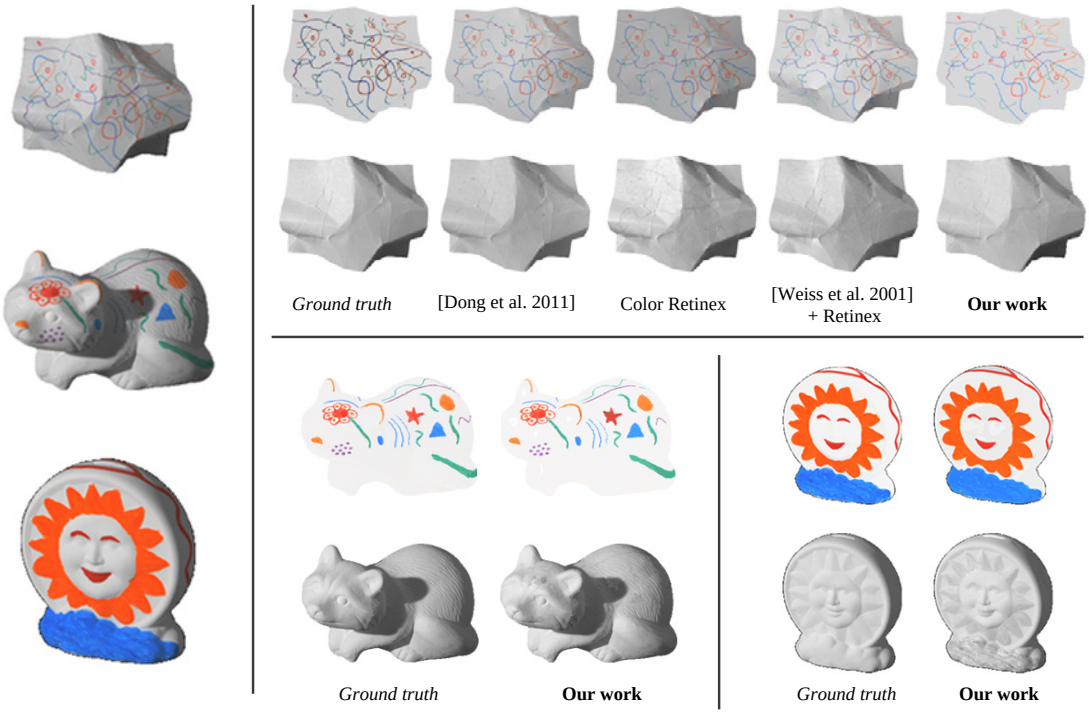


Figure 3.6: *Results using the MIT image dataset provided by Grosse et al. [GJAF09]. Left column: Input images (paper, raccoon and sun). Top row, comparison with previous works for paper. Bottom row: results for raccoon and sun compared to ground truth.*

Other MIT images are much more ill-posed for any intrinsic images algorithm, presenting extremely complex combinations of shading, smooth varying textures and poor chromaticity. Figure 3.7 shows our results for the case of *squirrel* (*deer* can be found in the Appendix A). On the top row, we show the original image along with our recovered shading and reflectance. Although our method yields reasonable results, some artifacts are clearly visible. This is mainly due to the extremely poor chromaticity variation of the original image, which hampers our segmentation step. The bottom row shows scatter plots of *squirrel*, along with *St. Basil* and *raccoon* for comparison purposes. Note how, despite its overall lack of chromaticity, *raccoon* presents clear distinct areas in the (a,b) plane, due to the scribbles painted on its surface. To the best of our knowledge, there is no published method that can successfully deal with images such as *squirrel* or *deer*.

Occlusion boundaries

Our method lets us handle a wide variety of images even in the presence of occlusion boundaries or sharp edges, where our assumptions do not hold. This is due to our robust linear system formulation: The number of equations describing C^0 continuity at an offending occlusion boundary is usually very small compared with the total number of equations in the system, which diminishes their influence in the result. For instance, the system solved for *St. Basil* is made up of 3557 equations, from which only 267 belong to occlusion boundaries (a mere 7%). In all the images shown in this work, we only found one

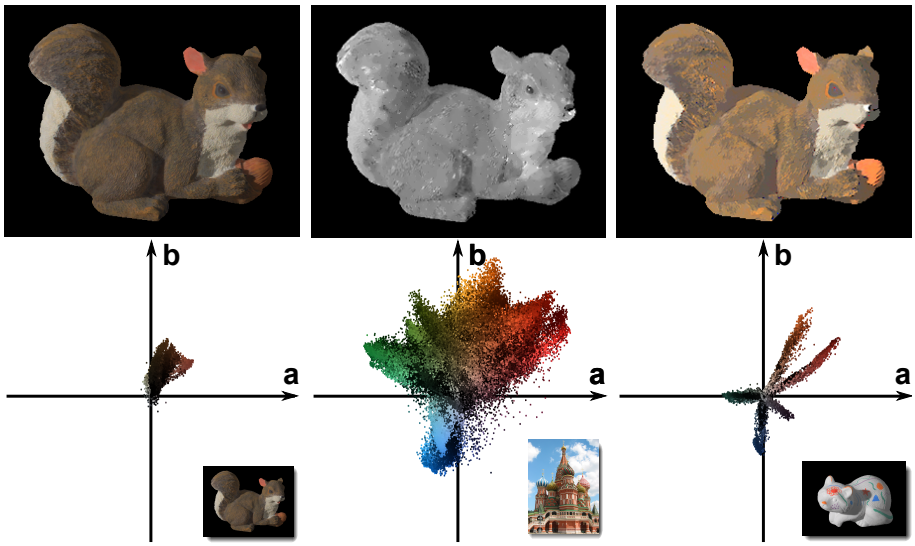


Figure 3.7: *A challenging case for our algorithm. Top row, from left to right: input image, and our intrinsic shading and reflectance. Notice the shadow close to the tail in the reflectance image. Bottom row: Comparison of scatter plots. From left to right: squirrel, St. Basil and raccoon. Notice the lack of chrominance variation in squirrel, compared to the other two plots.*

case (*doll*) where the percentage of offending equations was higher than 10% (48 over 282 equations, 17%). This image presents a unique combination of reflectance distributions that makes it especially challenging for our algorithm. It shows a very uniform background (which translates into very few clusters adding equations to the system), whereas the doll itself has lots of patches of different reflectance in contact with such background, due to the striped pattern of its clothes (adding a relatively high number of occlusion boundary equations). Note how, in contrast, *St. Basil* presents most of the colorful patches inside the building itself (few equations describing occlusion boundaries with the sky). Even though the automatic results provided by our system may be considered satisfactory, we can improve them by simply creating a matte of the doll, which can be easily done with existing image editing tools, and then solve the two resulting images (doll and background) as separate problems. Figure 3.8, bottom row, shows the result of directly applying our method, and the improved results with our quick matting profile created in *Photoshop*©. Artifacts due to the inherent difficulty of this image can be seen across all methods.

3.6 Conclusions

Decomposing an image into its intrinsic components is still an open problem with multiple potential applications. Automatic methods such as ours need to rely on reasonable assumptions or additional sources of information. On the other hand, existing user-assisted methods remain challenging for the average unskilled user, given the difficulty in telling apart the confounding factors of reflectance and shading in some situations. Our problem formulation is less restrictive than traditional Retinex-based methods, and allows us to relax our initial assumptions in certain cases. We have shown a wide variety of results, not only on natural scenes, but on the MIT dataset and even texture images as well. Ad-

ditionally, we have also provided a thorough comparison with previous approaches. Our algorithm produces better results than other automatic techniques on a broad range of input images, and on-par compared to user-assisted methods, but without the challenging task of providing the right strokes.

A potential line of future work would be to use our results as input to a simplified user interface, where it would be simpler to fix remaining artifacts. Also, our work could inspire and benefit from further research on segmentation methods. In conclusion, we believe that our approach offers an attractive trade-off between accuracy of the results, ease of use and efficiency.

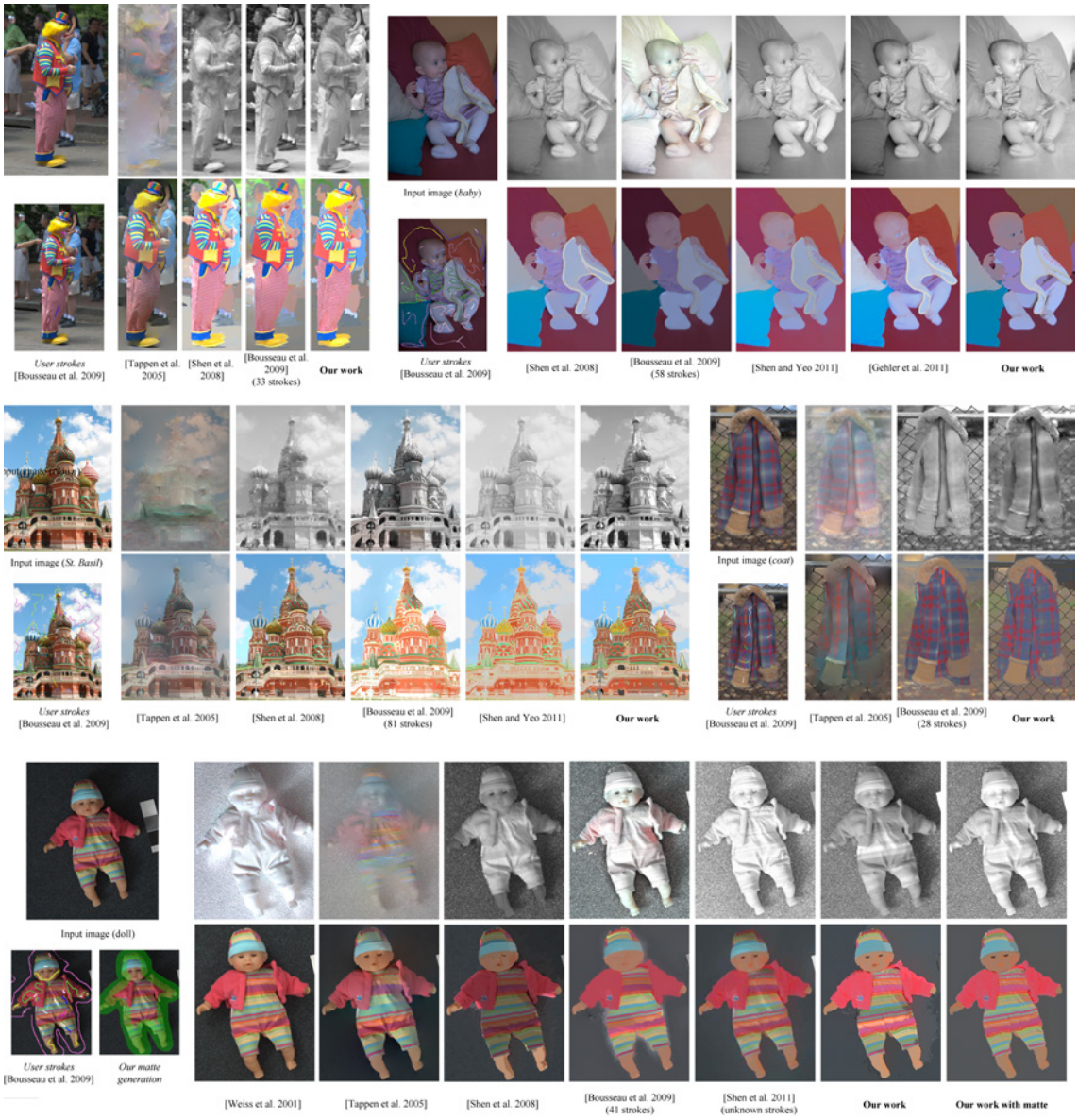


Figure 3.8: *Top left: clown. The methods by Shen et al. [STL08] and Tappen et al. [TFA05] introduce obvious artifacts in the shading. Our result is comparable with Bousseau et al. [BPD09] without requiring user strokes. Top right: baby. Our shading image is comparable with the one obtained by Bousseau et al. [BPD09] which needs 58 user strokes. Note that the result by Shen et al. [STL08] has quantized the leg of the baby while our result keeps the continuity in the shading. The shading by Shen and Yeo [SY11] is not totally homogeneous and contains reflectance information. Moreover, our reflectance image successfully captures the facial features, outperforming the other methods. Middle left: St. Basil. The methods by Shen et al. [STL08] and Tappen et al. [TFA05] introduce obvious artifacts in the shading. Bousseau et al.'s method produces great results, although it requires 81 user strokes (original image by Captain Chaos, flickr.com). Middle right: coat. Our result without any user strokes is not far from the result obtained by Bousseau et al. [BPD09] using 28 strokes. Bottom: doll. The automatic methods by Shen et al. [STL08] and Tappen et al. [TFA05] fail to obtain an homogeneous shading on the legs. See Appendix A for a bigger version of the images with more comparisons.*

Chapter 4

Intrinsic Image Decomposition by Co-linear Albedo

Assuming that objects have Lambertian surfaces and the incident light is monocromatic, similar to previous work [STL08, GRK+11, SY11, SYJL11], we formulate the intrinsic image decomposition problem as:

$$I = S \cdot R \tag{4.1}$$

where I corresponds to the input image, S is the single channel *illumination* component and R is the *reflectance* also named as *albedo* represented by a RGB color image. Our goal is to obtain S and R given I. In order to ensure instability and make the computations less complex, we take logarithms and transform equation 4.1 into the following,

$$i = s + r \tag{4.2}$$

where lowercase denotes the log quantities, e.g. $i = \ln(I)$.

Assumptions Our main assumption follows the classical Retinex based theories: we assume that shading is a smooth function. We agree that this assumption is a bit restrictive and only guarantees a solution for globally convex objects without occlusion boundaries or holes. However, for the scope of this thesis we focus on the solution for this particular case: globally convex objects or flat surfaces, and let the study of more complex scenes for future research.

4.1 Our approach: Co-linear Albedo

Based on the previous assumption, we can apply the gradient operator w.r.t. pixel coordinates obtaining,

$$\nabla i = \nabla s + \nabla r \tag{4.3}$$

Then, we subtract the previous equation in different color channels to cancel the shading variable, and observe that it exists an invariant direction $\mathbf{t}$ parallel to all the albedo

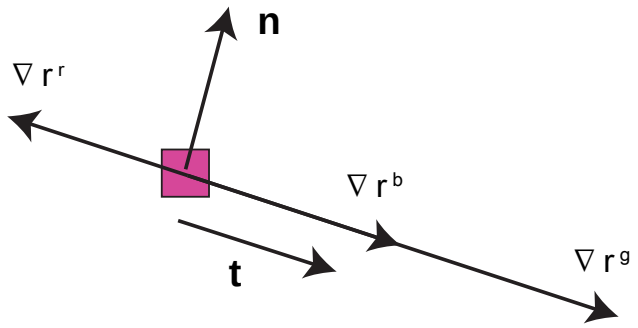


Figure 4.1: *Illustrative schema and notation used throughout this chapter. The albedo gradients* $\nabla r^r, \nabla r^g, \nabla r^b$ *at a pixel are suppose to be all parallel i.e. they lie along* $\hat{\mathbf{t}}$ *and are orthogonal to* $\hat{\mathbf{n}}$

gradients $\nabla r^r \parallel \nabla r^g \parallel \nabla r^b$ (see Figure 4.1). This vector $\mathbf{t}$, called the *tangent* vector, satisfies the following equation:

$$\mathbf{t} = \nabla i^\alpha - \nabla i^\beta = \nabla r^\alpha - \nabla r^\beta \quad \forall \alpha, \beta \in \{R, G, B\} \wedge \alpha \neq \beta \tag{4.4}$$

However, in practice, the log-albedo gradients may not be exactly parallel, so the tangent vector computed from, e.g., $\nabla r^r - \nabla r^g$ may not be the same as the one computed from $\nabla r^g - \nabla r^b$. Thus, in order to obtain an estimated invariant direction $\hat{\mathbf{t}}$, we compute a weighted average of the gradient albedo differences as follows,

$$\hat{\mathbf{t}} = \sum_{\alpha,\beta} \left(\nabla r^\alpha - \nabla r^\beta \right) = \sum_{\alpha,\beta} \mathbf{t}^{\alpha\beta} \quad \forall \alpha, \beta \in \{R, G, B\} \wedge \alpha \neq \beta \tag{4.5}$$

where we see the magnitude of the vector $\hat{\mathbf{t}}$ as a *confidence* measure which indicates if a tangent vector represents a true albedo direction. Thus, we define a confidence matrix $\mathcal{C} = |\hat{\mathbf{t}}|$ defined in the interval $[0 - 1]$. The confidence will be higher for gradients of the image which are really parallel, and lower (or zero) for gradients that are not. If the gradients cancel out and the direction is zero, we assume there is no change on the albedo. We can observe some of these confidence maps in Figure 4.2.

If we can find the tangent direction $\hat{\mathbf{t}}$, then we can compute a unitary normal vector $\hat{\mathbf{n}}$ perpendicular to it $\hat{\mathbf{n}} \perp \hat{\mathbf{t}}$, such that $\nabla r \, \hat{\mathbf{n}} = 0$.

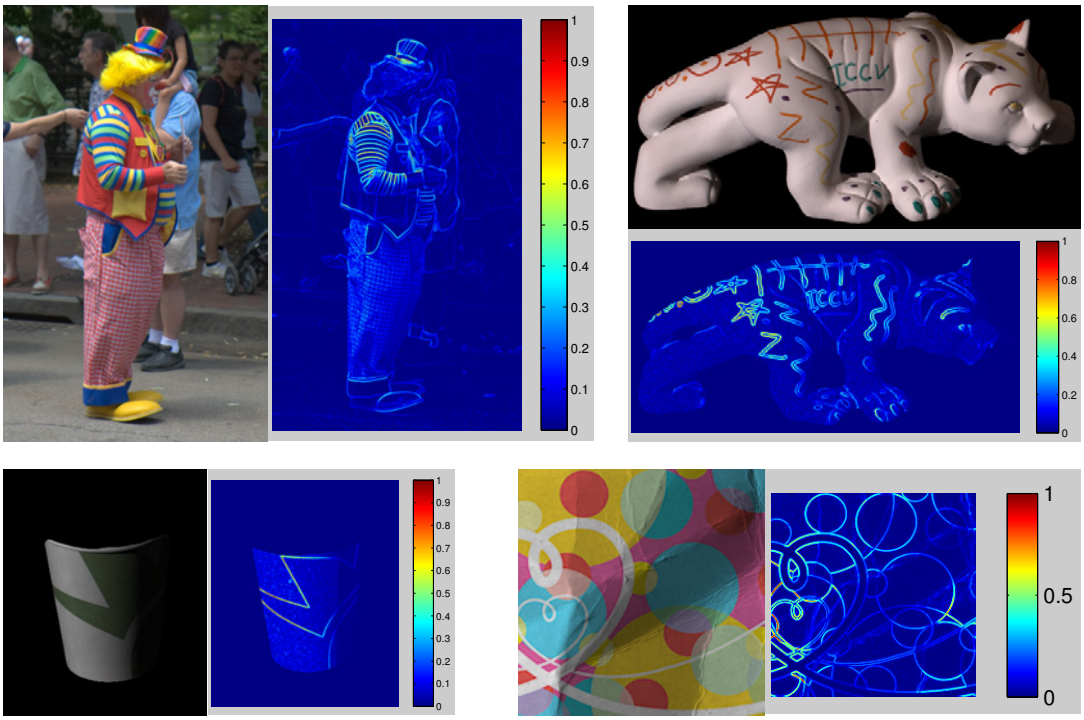


Figure 4.2: *Input images and their confidence maps. Blue-ish values mean lower confidence while red-ish mean a higher confidence. We observe that abrupt albedo changes usually imply higher confidence values. Notice that at deep shadows the confidence is lower. This happens because at darker areas the contrast is lower which results in less perceptible albedo changes.*

4.2 Optimization

Projecting Equation 4.3 into the normal vector $\hat{\mathbf{n}}$ gives,

$$\nabla i \cdot \hat{\mathbf{n}} - \nabla s \cdot \hat{\mathbf{n}} = 0 \tag{4.6}$$

Given i (the image) and knowing the approximated direction $\hat{\mathbf{n}}$, the task is to find s that satisfies Equation 4.6. Since it only constrains ∇s along the normal direction i.e. is rank deficient, and further $\hat{\mathbf{n}}$ can vanish, we use a regularization within and optimization-based framework to select a suitable solution. Thus, we define the following energy function to be minimized:

$$\mathrm{E} = w_d \mathrm{E}_{\mathrm{data}} + w_s \mathrm{E}_{\mathrm{smooth}} \tag{4.7}$$

Data Term

For the data term we use the Equation 4.6 multiplied by the confidence matrix $\mathcal{C}$.

$$\mathrm{E}_{\mathrm{data}} = \sum_p \mathcal{C}_p \left(\nabla i_p \cdot \hat{\mathbf{n}}_p - \nabla s_p \cdot \hat{\mathbf{n}}_p\right)^2 \tag{4.8}$$

This confidence matrix $\mathcal{C}$ defined in the range $[0-1]$ has higher values for pixels whose gradients are actually parallel and thus, they satisfy the Equation 4.6. And very low values, almost zero, for those points which not. In such case, the resulting product will tend

to zero which means that the point will not contribute to specify the solution.

Intuitively, this data term captures the shading variations at the boundaries where albedo changes (high confidence), but does not affect the rest of the points where the confidence is low. For this reason, a regularizer term is necessary to obtain an univocal solution.

Smoothness Term

The goal of this smoothness term is to avoid the null solution since the data term is rank deficient. Hence, we use the Laplacian operator to guarantee a smooth shading function:

$$\mathrm{E}_{\mathrm{smooth}} = \sum_p \left(\Delta \mathcal{S}_p\right)^2 = \sum_p \left(\nabla^2 \mathcal{S}_p\right)^2 \tag{4.9}$$

In theory, this operator must be applied over the original shading domain (not log-domain) which is linear. However, by doing so, we would have a nonlinear energy function which is harder to optimize. Therefore, we transform the original smoothness term into the following:

$$\mathrm{E}_{\mathrm{smooth}} = \sum_p \left(\Delta s_p\right)^2 = \sum_p \left(\nabla^2 s_p\right)^2 \tag{4.10}$$

where the only change is that we use the logarithm of the shading instead of the shading itself. The effect of this change over the resulting solution is that it produces shading images more darker and with higher contrast than the ground truth.

To minimize this error, we reduce the influence of this term by multiplying it by a weighting matrix $\mathcal{W}$. This new operator is commonly known as the *weighted Laplacian*:

$$\mathrm{E}_{\mathrm{smooth}} = \sum_p \mathcal{W}_p \left(\Delta s_p\right)^2 = \sum_p \mathcal{W}_p \left(\nabla^2 s_p\right)^2 \tag{4.11}$$

This matrix $\mathcal{W}$ depends on the confidence matrix $\mathcal{C}$ computed from Equation 4.5 and is defined by:

$$\mathcal{W} = \exp\left(\frac{-1}{\left(\nabla \mathcal{C}\right)^2 + \epsilon}\right) \tag{4.12}$$

where $\epsilon = 0.15$ guarantees that the weight is always greater than zero and has a meaningful value. Due to image noise or other artifacts, the computed confidence matrix may not be very accurate at lower values. Thus, we use this function as a soft threshold to enforce low confidence values to be zero and to make the optimization more stable. The behavior of the function is shown in Figure 4.3.

In Figure 4.4 we can see the resulting shading and reflectance images with the normal Laplacian (Equation 4.10) and with the weighted Laplacian (Equation 4.11). As we can see, if all the pixels are weighted similarly using equation 4.10, the non-linearity of the log of the shading produces too dark shading images and also over-smooth the solution (Figure 4.4,b) . Although perceptually the shading seems correct, observing the resulting reflectance the error appears noticeable.

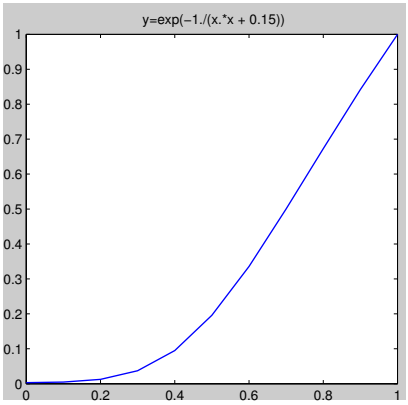


Figure 4.3: *Weighting function of Equation 4.12 used to enforce confidence gradients values to be zero. Notice that the function is almost zero for values smaller than 0.2 but for greater values it grows fast.*

Solution

In order to solve for the unknown values of s, we just have to transform it into a matrices system and solve it by an unconstrained least squares optimization. The constants w_s and w_d have different values depending on the images, although in general, for solutions using Equation 4.10 they are $w_d = 1000$ and $w_s = 0.1$, while for solutions using Equation 4.11 they are $w_d = 1$ and $w_s = 1000$. Once we have s computed, the shading is obtained from $S = e^s$ and the reflectance is obtained by simple division $R = I/S$.

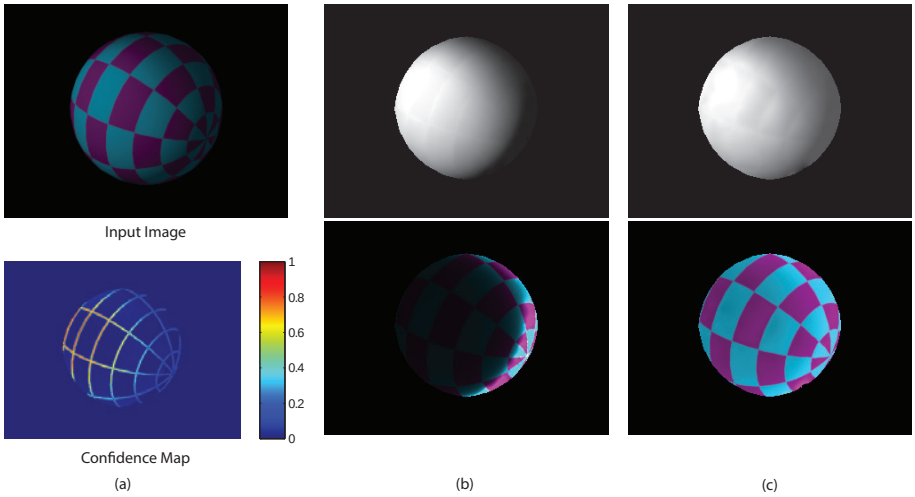


Figure 4.4: *Effect of the Laplacian operator over the output solution. (a) Input image and confidence map.(b) Shading and reflectance obtained with Equation 4.10. (c) Shading and reflectance obtained with Equation 4.11. Notice how the shading produced by the non-weighted Laplacian (b) is two dark in the shadow region due to the non-linearity of the logarithm function, which results in an incorrect reflectance image.*

4.3 Results

We show several examples for which the formulation works well and at very low rates of computation (approx. 1 second). We have tested the method mainly with globally convex objects and simple scenes where the foreground object has been masked out to avoid occlusions. However, some examples with more complex scenes are also presented. In Figures 4.5 and 4.6 we show the shading and reflectance obtained with our algorithm for different scenes.

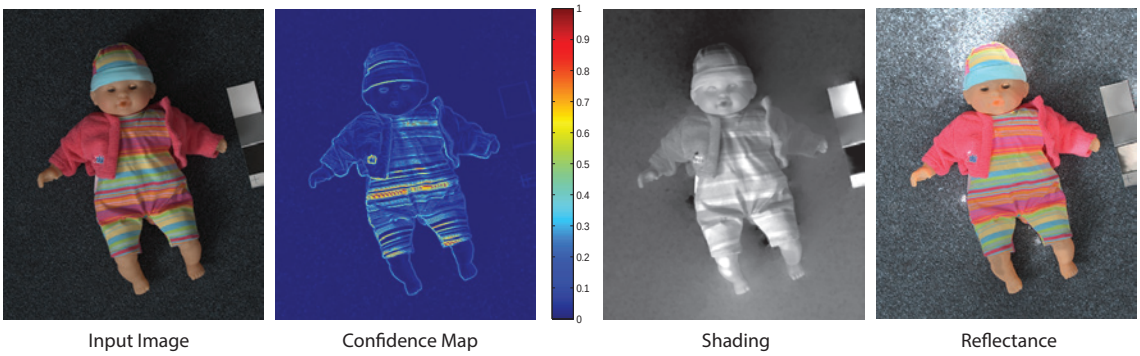


Figure 4.5: *Intrinsic images obtained with our algorithm for* doll. *Notice that those regions with higher confidence values are better decomposed. Also, observe that the eyes of the doll have been correctly identified as reflectance while most of the other methods fail at this case.*

Figure 4.6a-c, shows some examples of simple objects and flat surfaces. Our decompositiosn are quite accurate, in particular, Figure 4.6,a compares our results with the ground truth proving our solution to be very similar to it.

We also show examples of more complex scenes where the foreground object has not been masked out (see Figures 4.5 and 4.6,d). We notice that while the main object is well approximated, the rest of the scene has certain errors. This problem arises from our assumption of smooth shading and can be solved if we separate foreground from background and handle each part individually.

Although the shading image is perceptually correct in several cases, if we observe in detail the reflectance images of Figure 4.6,c, we can still notice traces of shading. The reason is twofold: first, we do not impose constraints to the reflectance component, contrarily to other methods which assume it to be piecewise constant; and second, the aforementioned non-linearity of the log of the shading yields to incorrect contrasted shading, error which propagates to the reflectance image.

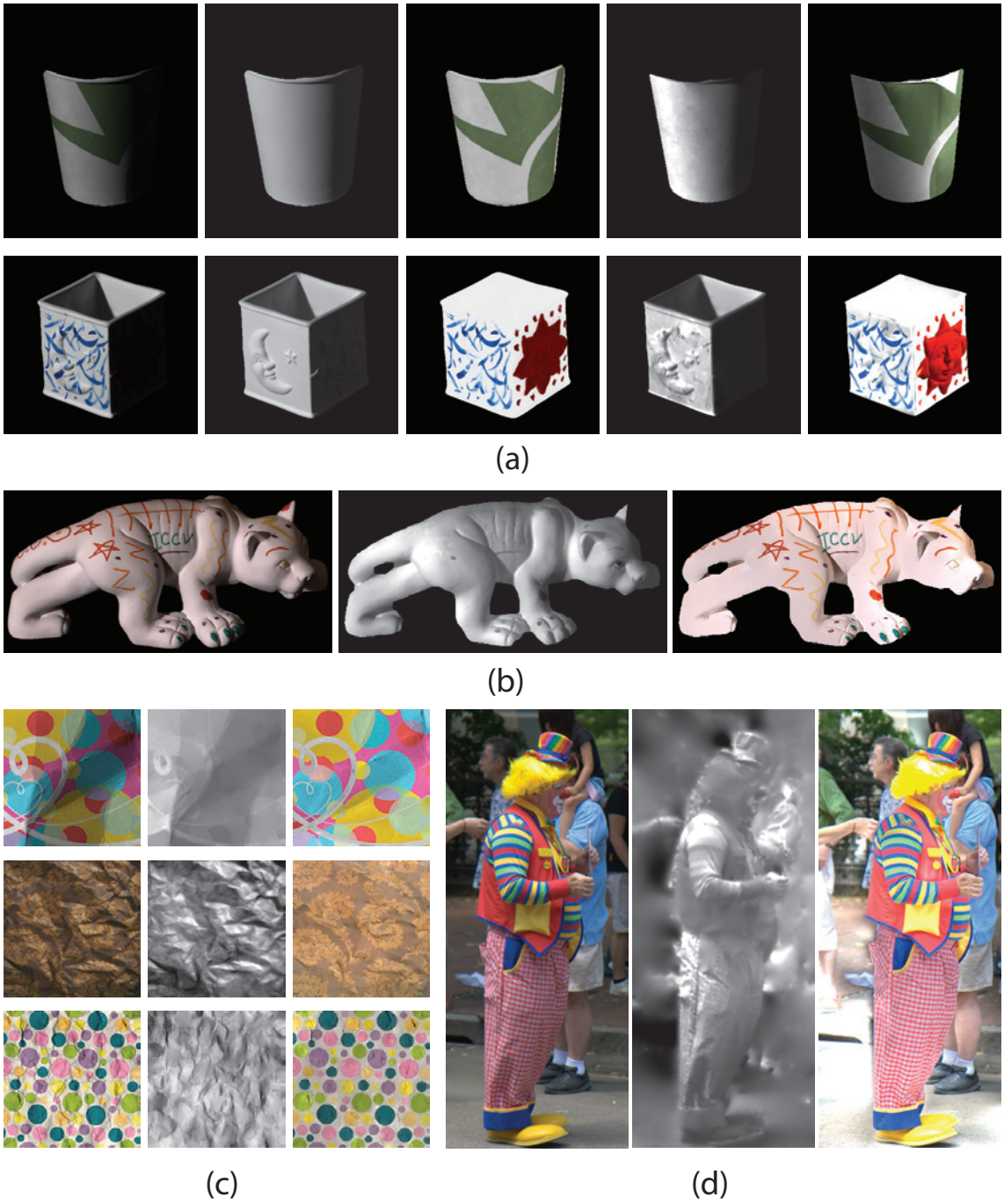


Figure 4.6: *Intrinsic images obtained with the algorithm. (a) First column: input image; Second-Third columns: Ground truth shading and reflectance; Last columns: our shading and reflectance. (b),(c),(d) Input image, shading and reflectance obtained with our algorithm.*

4.4 Conclusions and future work

In this work, we have explored the parallelism between the gradients of the log albedos to obtain the intrinsic shading and reflectance. We have tested the method on globally convex objects, flat surfaces and more complex scenes. The experiments performed so far suggest that this algorithm has very high potential, and it opens the door to interesting future work.

In order to reduce remaining shading at the reflectance component, an interesting idea would be to include a new energy term which constrain the reflectance to be piecewise constant like previous work [SYJL11]. Then, we could build an optimization function which solve for both reflectance and shading at the same time.

Another idea would be to extend the assumption of parallelism from a 2D line to a plane in 3D. In such case, special attention needs to be pay to compare the potential solution with the work by Bousseau et al. [BPD09], that assume that changes in albedo lie in a local 3D plane in RGB.

The logical extension would be to include minimum user input or an interactive application which automatically detect and inform about ambiguous regions of the image so that the user can easily correct it.

Chapter 5

Conclusions and future work

Decomposing an image into its intrinsic components is still an open problem with multiple potential applications. Automatic methods need to rely on reasonable assumptions or additional sources of information. On the other hand, existing user-assisted methods remain challenging for the average unskilled user, given the difficulty in telling apart the confounding factors of reflectance and shading in some situations. In this master thesis we propose two different methods to solve the problem of intrinsic image decomposition from a single image without user interaction.

Our first solution, *Intrinsic Images by Clustering*, follows the classical Retinex-based assumptions although we relax them to allow a wider range of input scenarios. This method has been proved to produce better results than other automatic techniques and on-par with user-assisted methods, but without the difficult task of providing the correct strokes.

Although the first method proposed and published performs well for several input images, the ill-possedness of the problem makes it impossible to obtain always an univocal solution. Hence, for certain input images with smoothly varying textures, the segmentation is not able to capture the fine grain of the reflectance yielding poor results. For that reason, we explored a second solution which works at pixel level instead of at cluster level, which we believed to have a great potential for solving such scenes.

The second solution, *Intrinsic Images by Co-linear Albedo*, is a formulation currently in developing. We have performed an initial evaluation obtaining compelling results for the images tested. The obtained results and conclusions opens the door for interesting future work.

As far as I am concerned, my work in the field of intrinsic image decomposition ends with this master thesis. However, there is a lot of potential work that can be done to continue with this research. An interesting idea would be to combine both methods in a multi-level optimization framework. This way, we could take the best of both methods in a unified solution.

After the deep study of the problem accomplished in this master thesis, I conclude that in order to obtain perfect accurate decompositions some user input will always be necessary due to the inherent ambiguity of the problem. Hence, the next step should be to seek a solution which requires the least possible user interaction.

Bibliography

[BBC+94] R. Barret, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, R. Pozo, V. Eijkhout, H. Van der Vorst, and C. Romine. *Templates for the Solution of Linear Systems: Building Blocks for iterative Methods, 2nd Edition*. SIAM, 1994.

[BPD09] Adrien Bousseau, Sylvain Paris, and Frédo Durand. User assisted intrinsic images. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2009)*, 28(5), 2009.

[BT78] H.G. Barrow and J.M. Tenenbaum. Recovering intrinsic scene characteristics from images. *Computer Vision Systems*, pages 3–26, 1978.

[DTPG11] Yue Dong, Xin Tong, Fabio Pellacini, and Baining Guo. AppGen : Interactive Material Modeling from a Single Image. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2011)*, (2), 2011.

[FDB92] Brian V. Funt, Mark S. Drew, and Michael Brockington. Recovering shading from color images. In *ECCV-92: Second European Conference on Computer Vision*, pages 124–132. Springer-Verlag, 1992.

[FH04] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59:2004, 2004.

[FS95] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*, pages 23–37. Springer-Verlag, 1995.

[GDFL04] Mark S. Drew Graham D. Finlayson and Cheng Lu. Intrinsic images by entropy minimization. In *Proc. 8th European Conf. on Computer Vision, Praque*, pages 582–595, 2004.

[GGLM11] Elena Garces, Diego Gutierrez, and Jorge Lopez-Moreno. Graph-based reflectance segmentation. In *Proceedings of SIACG 2011*, 2011.

[GJAF09] 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.

[GRK+11] Peter V. Gehler, Carsten Rother, Martin Kiefel, Lumin Zhang, and Bernhard Schölkopf. Recovering intrinsic images with a global sparsity prior on reflectance. In *NIPS*, page 765, 2011.

[Hor74] B. K. Horn. Determining lightness from an image. *Computer Graphics and Image Processing*, 3(4):277–299, December 1974.

[JSW10] Xiaoyue Jiang, Andrew J. Schofield, and Jeremy L. Wyatt. Correlation-based intrinsic image extraction from a single image. In *Proceedings of the 11th European conference on Computer vision: Part IV*, ECCV'10, pages 58–71, Berlin, Heidelberg, 2010. Springer-Verlag.

[KES+03] Ron Kimmel, Michael Elad, Doron Shaked, Renato Keshet, and Irwin Sobel. A variational framework for retinex. *International Journal of Computer Vision*, 52:7–23, 2003.

[KMN+04] T. Kanungo, D. M. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Y. Wu. A local search approximation algorithm for k-means clustering. *Computational Geometry: Theory and Applications*, 28:89–112, 2004.

[LBD11] Pierre-Yves Laffont, Adrien Bousseau, and George Drettakis. Rich Intrinsic Image Separation for Multi-View Outdoor Scenes. Research Report RR-7851, INRIA, December 2011.

[LM71] E. H. Land and J. J. McCann. Lightness and retinex theory. *Journal of the Optical Society of America*, 61(1), 1971.

[LMHRG10] Jorge Lopez-Moreno, Sunil Hadap, Erik Reinhard, and Diego Gutierrez. Compositing images through light source detection. *Computers & Graphics*, 34(6):698–707, 2010.

[LWQ+08] Xiaopei Liu, Liang Wan, Yingge Qu, Tien-Tsin Wong, Stephen Lin, Chi-Sing Leung, and Pheng-Ann Heng. Intrinsic colorization. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2008)*, pages 1–9, 2008.

[OW04] Ido Omer and Michael Werman. Color lines: image specific color representation. In *Conference on Computer vision and pattern recognition*, CVPR'04, pages 946–953. IEEE Computer Society, 2004.

[STL08] Li Shen, Ping Tan, and Stephen Lin. Intrinsic image decomposition with non-local texture cues. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:1–7, 2008.

[SY11] Li Shen and Chuohao Yeo. Intrinsic images decomposition using a local and global sparse representation of reflectance. In *Computer Vision and Pattern Recognition*, pages 697–704. IEEE, 2011.

[SYJL11] Jianbing Shen, Xiaoshan Yang, Yunde Jia, and X. Li. Intrinsic images using optimization. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3481–3487. IEEE, 2011.

[TAF06] Marshall F. Tappen, Edward H. Adelson, and William T. Freeman. Estimating intrinsic component images using non-linear regression. In *Conference on Computer Vision and Pattern Recognition - Volume 2*, CVPR '06, pages 1992–1999. IEEE Computer Society, 2006.

[TFA05] 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*, 27:1459–1472, 2005.

[Wei01] Yair Weiss. Deriving intrinsic images from image sequences. *Computer Vision, IEEE International Conference on*, 2:68, 2001.

Universidad Zaragoza
1542

Trabajo Fin de Máster

Practical Intrinsic Image Decomposition

Autora

Elena Garcés García

Director

Diego Gutiérrez Pérez

Escuela de Ingeniería y Arquitectura

Septiembre 2012

Appendix

Appendix A

Extended results

Figures A.1 to A.9 (From left to right) First column, input image and scatter plot of pixel data in the (a,b) plane (Lab color space). Second column, k-means segmentation according to (a,b) pixel coordinates; third column, final clustering yielded by our method taking into account spatial information (both, second and third rows, are depicted in false color). Last columns, the resulting shading and reflectance intrinsic images.

Figure A.10 Our intrinsic shading and reflectance for an image inspired by Dong et al. 2011. Our technique can also be used in conjunction with others, such as the image-based material modeling technique recently presented by Dong et al. 2011

Figures A.11 to A.15 Comparison with the state of the art methods in intrinsic image decomposition.

Figures A.16 to A.17 Intrinsic images for the MIT dataset. First column, input image. Second and third columns, ground truth shading and reflectance. Last columns, our resulting shading and reflectance. The gamma of the images has been corrected for visualization purposes.

Figure A.1: *Lollipop (original image by Thalita Carvalho, flickr.com)*

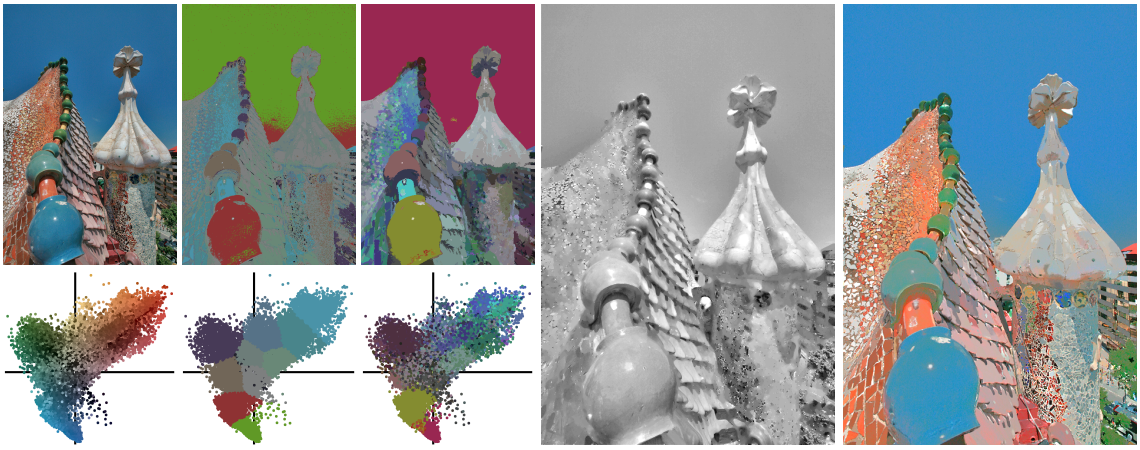


Figure A.2: *Batlló house (original image by lukasz dzierzanowski, flickr.com)*

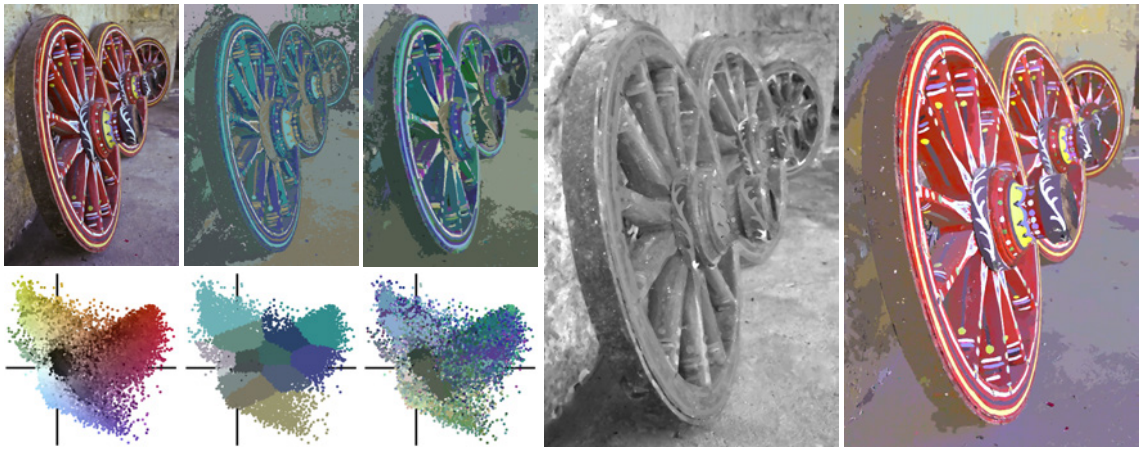


Figure A.3: *Wheels (original image by Angela Smith Kirkman)*

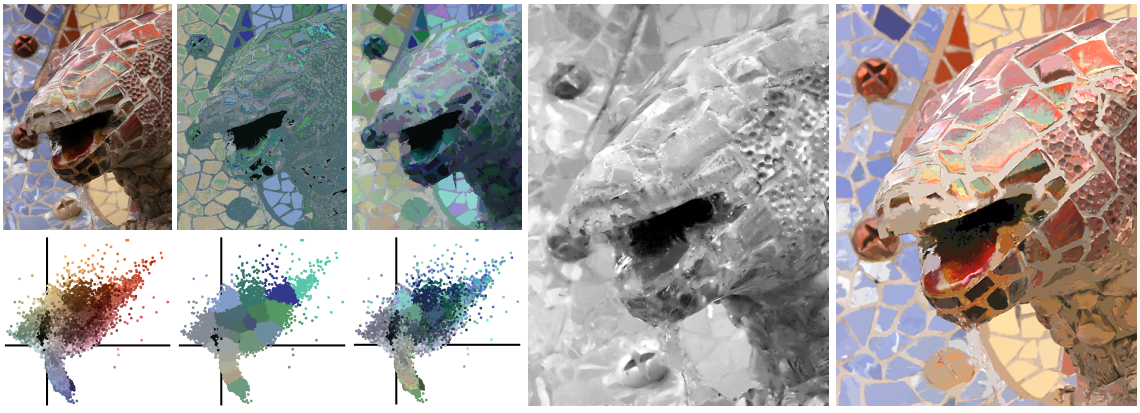


Figure A.4: *Dragon (original image by Jordanhill School D&T Dept, flickr.com)*

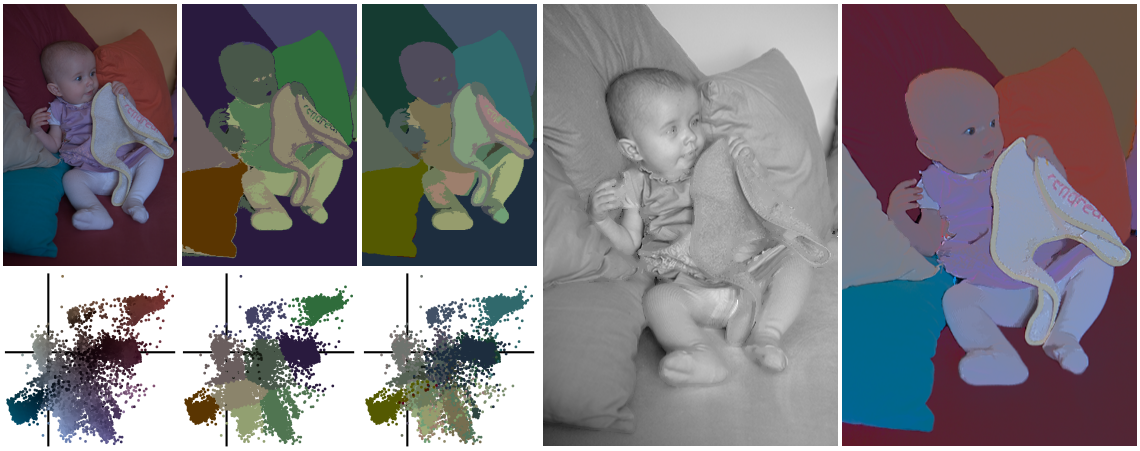


Figure A.5: *Baby*

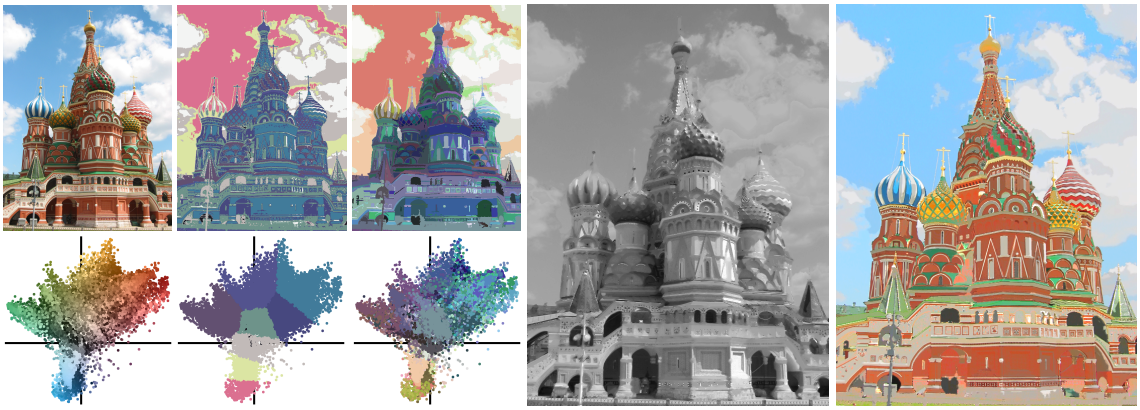


Figure A.6: *St. Basil (original image by Captain Chaos, flickr.com)*

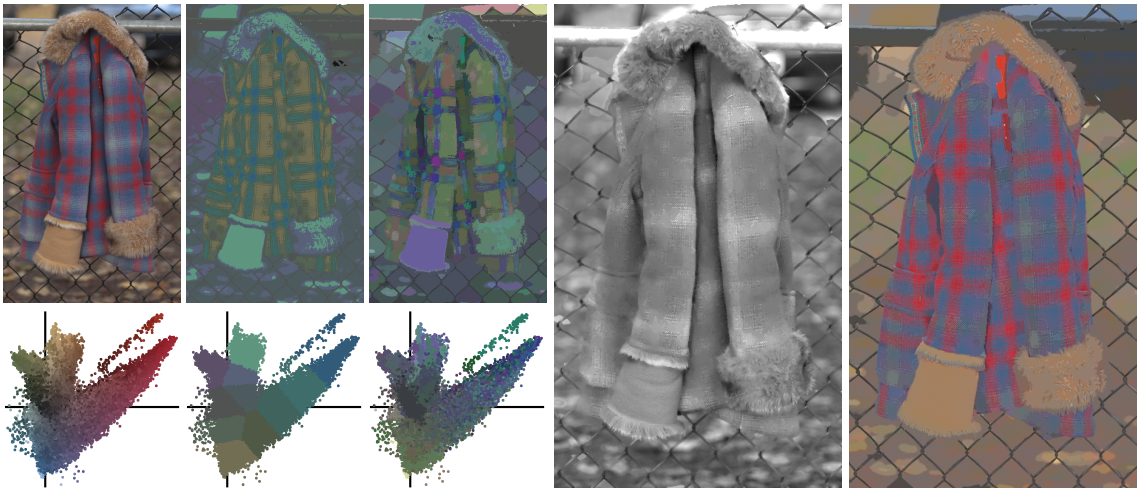


Figure A.7: *Coat*

Figure A.8: *Clown*

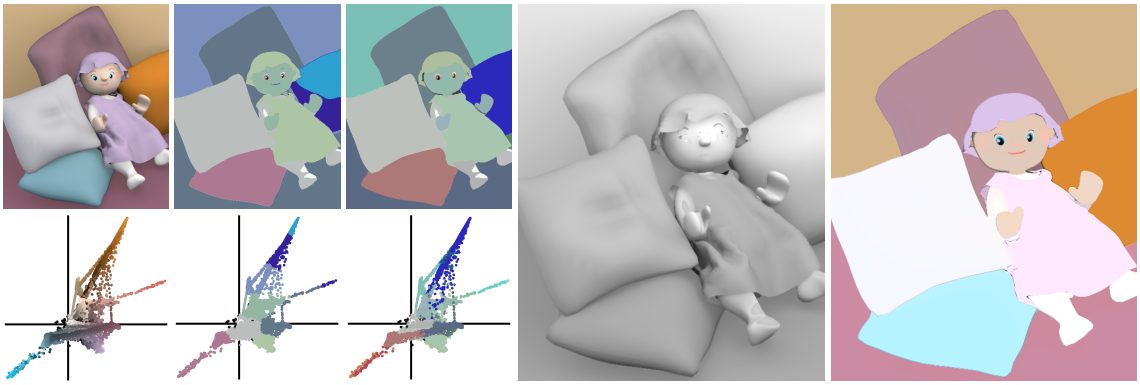


Figure A.9: *Synthetic*

Figure A.10: *Left: Input texture image. Middle: Our intrinsic reflectance. Right: Our intrinsic shading.*

Figure A.11: *Baby and St. Basil (original image by Captain Chaos, flickr.com)*


Input image (*clown*)
User strokes [Bousseau et al. 2009]
[Tappen et al. 2005]
[Shen et al. 2008]
[Bousseau et al. 2009] (33 strokes)
[Shen et al. 2011] (auto)
Our work

Figure A.12

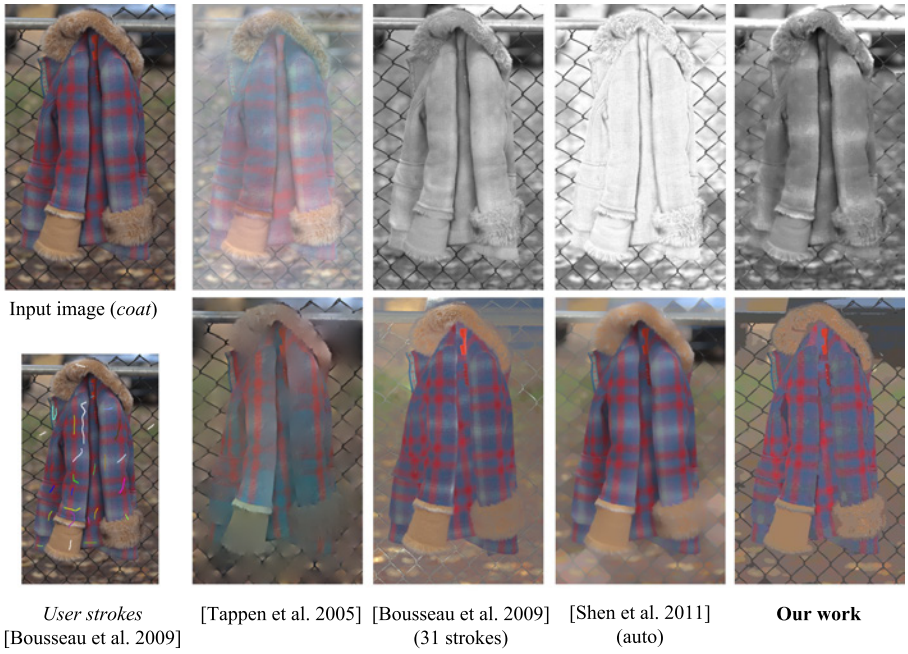


Figure A.13

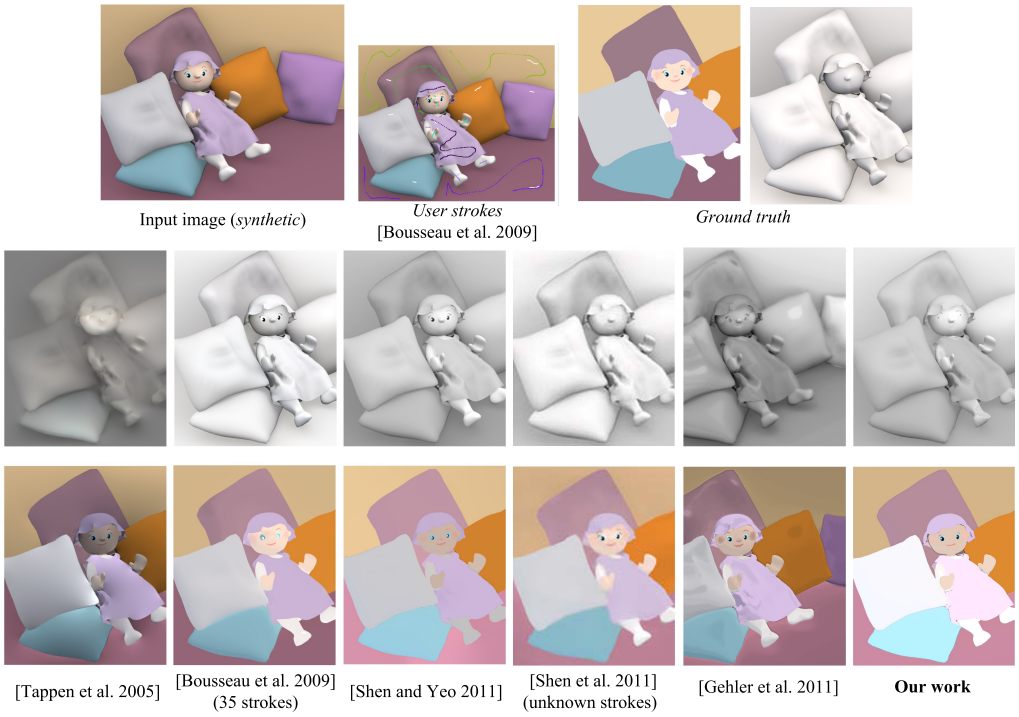


Figure A.14

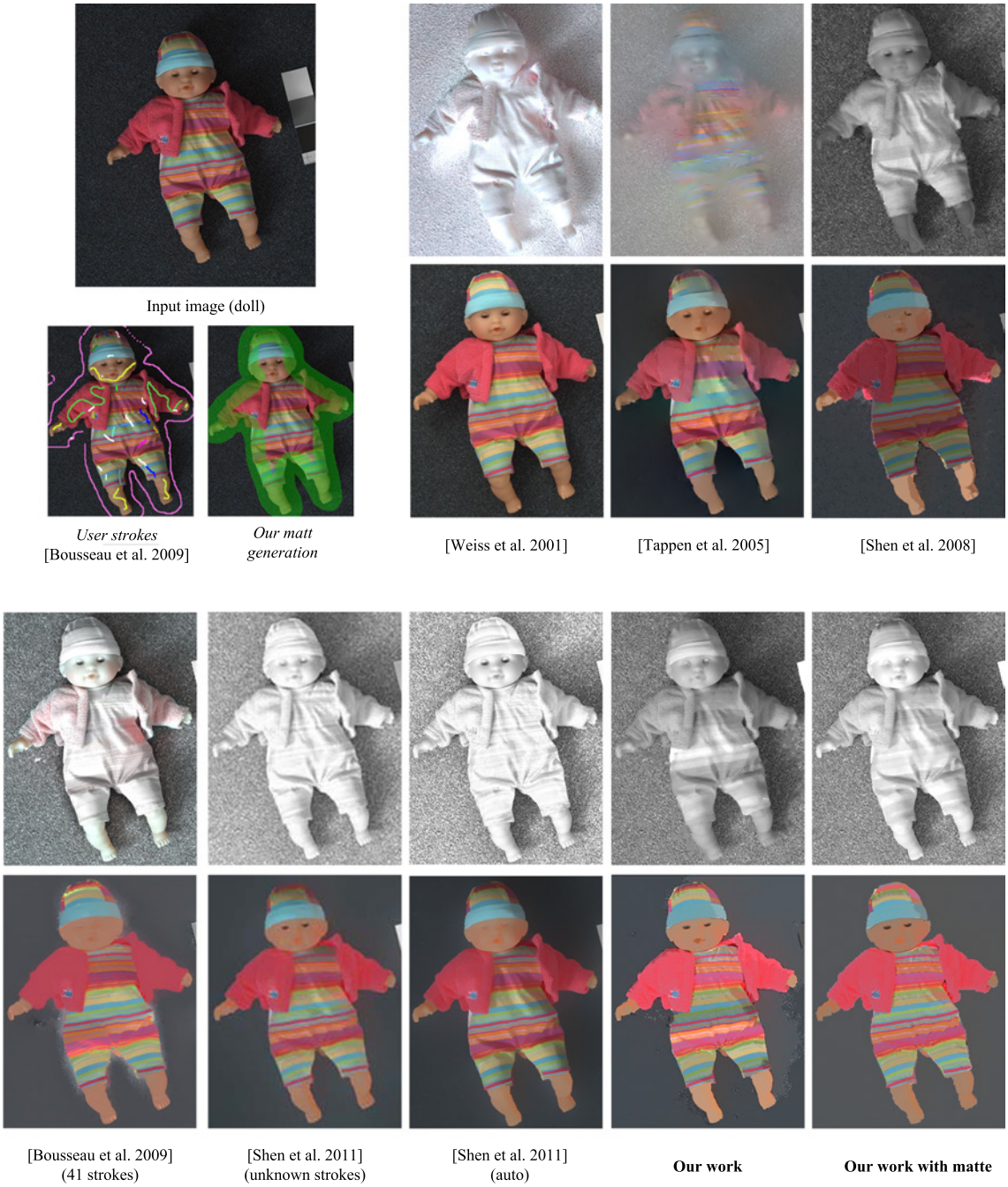

Input image (doll)
User strokes
[Bousseau et al. 2009]
Our matt
generation
[Weiss et al. 2001]
[Tappen et al. 2005]
[Shen et al. 2008]
[Bousseau et al. 2009]
(41 strokes)
[Shen et al. 2011]
(unknown strokes)
[Shen et al. 2011]
(auto)
Our work
Our work with matte

Figure A.15

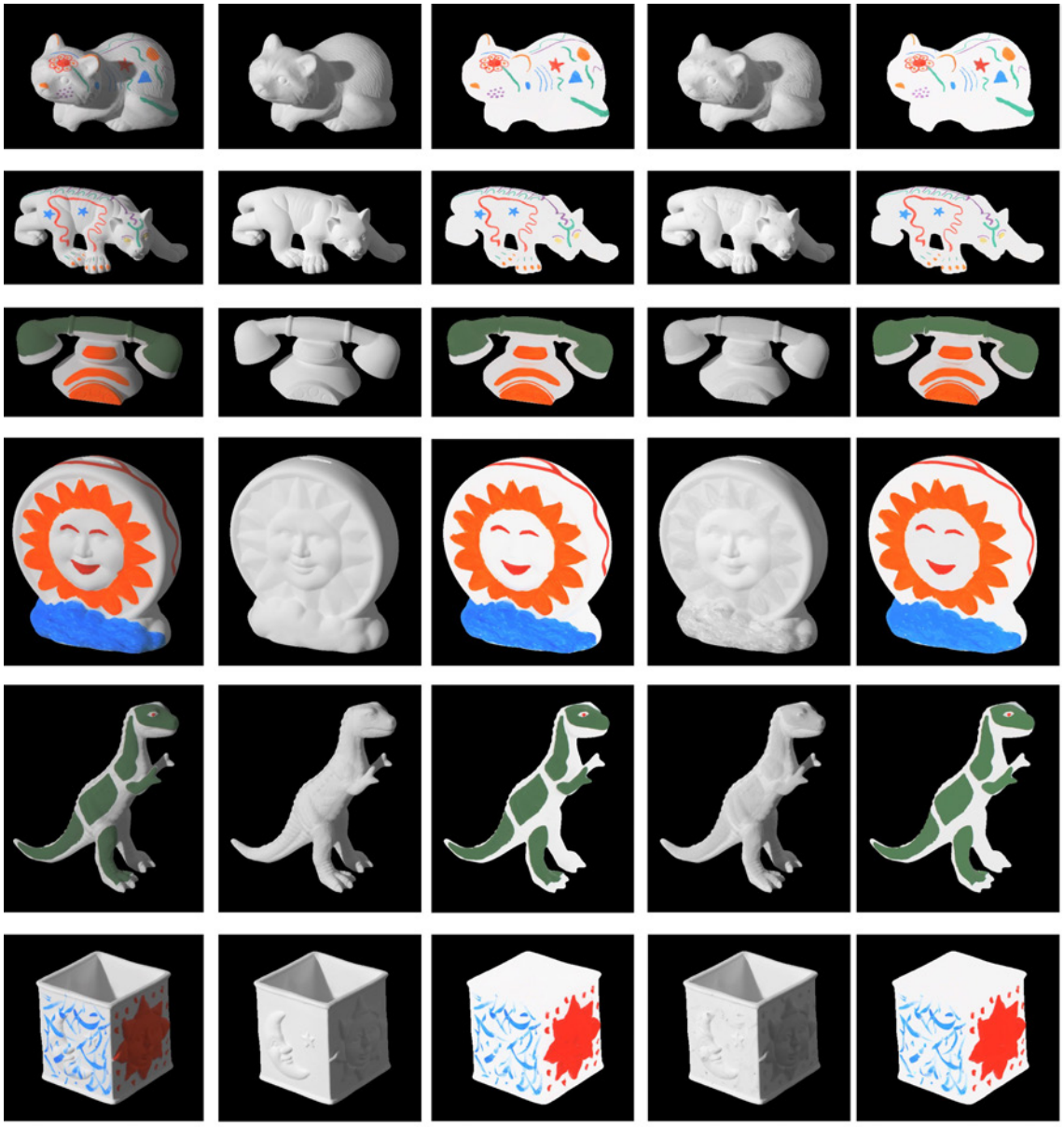


Figure A.16