SUN POSITION ESTIMATION ON TIME-LAPSE VIDEOS FOR AUGMENTED REALITY APPLICATIONS

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE DEGREE OF

MASTER OF SCIENCE

IN

COMPUTER ENGINEERING

By

Hasan Balcı

July, 2015

SUN POSITION ESTIMATION ON TIME-LAPSE VIDEOS FOR
AUGMENTED REALITY APPLICATIONS

By Hasan Balcı

July, 2015

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

_____
Prof. Dr. Uğur Güdükbay (Advisor)

_____
Prof. Dr. Özgür Ulusoy

_____
Asst. Prof. Dr. Erkut Erdem

Approved for the Graduate School of Engineering and Science:

_____
Prof. Dr. Levent Onural
Director of the Graduate School

ABSTRACT

SUN POSITION ESTIMATION ON TIME-LAPSE VIDEOS FOR AUGMENTED REALITY APPLICATIONS

Hasan Balcı

M.S. in Computer Engineering

Advisor: Prof. Dr. Uğur Güdükbay

July, 2015

Realistic illumination of virtual objects on Augmented Reality (AR) environments is important in terms of achieving visual coherence. This thesis proposes a novel approach that facilitates the illumination estimation on time-lapse videos and gives the opportunity to combine AR technology with time-lapse videos in a visually consistent way. The proposed approach works for both outdoor and indoor environments where the main light source is the Sun. We first modify an existing illumination estimation method that aims to obtain sparse radiance map of the environment in order to estimate the initial Sun position. We then track the hard ground shadows on the time-lapse video by using an energy-based pixel-wise method. The proposed method aims to track the shadows by utilizing the energy values of the pixels that forms them. We tested the method on various time-lapse videos recorded in outdoor and indoor environments and obtained successful results.

Keywords: sun position estimation, light source position estimation, illumination estimation, time-lapse video, shadow tracking, augmented reality.

ÖZET

ARTIRILMIŞ GERÇEKLİK UYGULAMALARI İÇİN HIZLANDIRILMIŞ ÇEKİM VİDEOLARDA GÜNEŞ POZİSYONU TAHMİNİ

Hasan Balcı
Bilgisayar Mühendisliği, Yüksek Lisans
Tez Danışmanı: Prof. Dr. Uğur Güdükbay
Temmuz, 2015

Artırılmış Gerçeklik (AG) ortamlarındaki sanal nesnelerin gerçekçi bir şekilde aydınlatılması görsel uyumluluğun sağlanması açısından önemlidir. Bu tez hızlandırılmş çekim videolarda aydınlatma tahminini kolaylaştıran özgün bir yaklaşım önermekte ve AG teknolojisiyle hızlandırılmş çekim videoların görsel olarak uyumlu bir şekilde birleşmesine olanak sağlamaktadır. Önerilen yaklaşım ana ışık kaynağının Güneş olduğu iç ve dış ortamlarda çalışmaktadır. Bu yaklaşımda, ilk olarak ortamın aralıklı ışınım haritasını çıkarmaya alışan mevcut bir aydınlatma tahmin yöntemi kullanılarak Güneş'in ilk pozisyonunu tahmin edilir. Daha sonra hızlandırılmış çekim videodaki sert yer gölgeleri enerjiye dayalı piksel bazlı bir yöntem kullanılarak takip edilir. Bu yöntem gölgeleri oluşturan piksellerin enerji değerlerini gölgeleri takip etmek için kullanmayı amaçlar. Önerilen yöntem iç ve dış ortamlarda çekilmiş çeşitli hızlandırılmş çekim videolarda denenmiş ve başarılı sonuçlar elde edilmiştir.

Anahtar sözcükler: güneş pozisyon tahmini, ışık kaynaği pozisyon tahmini, ışıklandırma tahmini, hızlı-ekim video, gölge takibi, artırılmış gerçeklik.

Acknowledgement

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation and Scope

Augmented Reality (AR) is a technology that combines real world environment with virtual entities such as graphics, video, sound or haptic feedback. Current technology allows people to experience AR technology by using a variety of devices from head-mounted displays to mobile phones and tablets. With the benefits it provides, AR has a broad application area, including medicine, education, commerce, advertising and entertainment [1].

One of the main goals in AR technology is to achieve seamless integration of the virtual objects into the real environment, especially in a visual context, so that the user cannot differentiate the virtual objects from the real ones. To this end, consistent illumination of virtual objects within the real environment is important to obtain visual coherence. This can be achieved in different ways, such as estimating the locations of the light sources in the real environment, considering the shadows cast by/on virtual objects or calculating the global illumination.

Light source estimation in indoor environments is more complex than on the outdoor environments, since indoor environments may include a lot of different

kinds of light sources while the main light source on the outdoor environments is the Sun. However, the light sources in indoor environments are generally static, whereas the position of the Sun changes over time. Regarding the environments that are affected mainly by Sun, an AR user may want to observe the virtual objects on different times of the day where the Sun is in a different position each time. For example, an architect may want to analyse the appearance of a building in the course of the day. For this purpose, time-lapse videos can be formed by capturing frames of the real environment during the desired period of the day and then they can be combined with AR technology.

We introduce a new approach that provides visual consistency on AR environments integrated on time-lapse videos where the main light source is the Sun. First, we estimate the initial position of the Sun from the first frame of the time-lapse video by using a modified and enhanced state-of-the-art method. Then, by keeping track of the shadow length and direction found on the ground, we estimate the change in the Sun position and direction on each frame and adjusts the illumination of the virtual objects accordingly.

1.2   Contributions

This thesis has mainly two contributions:

- We modify and enhance an existing method, which tries to estimate the illumination of a real scene from a single image, especially for the purpose of estimating the Sun position.

- We propose a new algorithm that can estimate the Sun position fast and accurately on time-lapse videos. By means of the energy-based pixel-wise algorithm we developed, we can track the hard ground shadows during the time-lapse video and estimate the changes in the Sun position easily.

1.3 Thesis Organization

The rest of this thesis is divided into five chapters. Chapter 2 summarizes the related studies about illumination estimation for different environments and purposes. Chapter 3 explains the existing illumination estimation method which we use in this study and how we modified and improved it especially to estimate the initial Sun position. Chapter 4 provides the details of the energy-based pixel-wise algorithm we developed for tracking the hard ground shadows in order to estimate the changes in the Sun position during the time-lapse video. Chapter 5 present the results of our study by evaluating them both qualitatively and quantitatively. Chapter 6 concludes by discussing the advantages and drawbacks of the proposed approach together with possible future extensions.

Chapter 2

Background and Related Work

There are many studies related to illumination estimation on AR environments, whereas the number of studies which combines the illumination estimation with time-lapse videos is limited. The studies on illumination estimation on AR environments are intended for outdoor environments, indoor environments, mobile devices, or other specific purposes. We first summarize the research on illumination estimation on AR environments in different categories defined according to the intended fields. We then discuss the studies that combine illumination estimation with time-lapse videos. We finally describe the existing illumination estimation method that is modified and used as a basis of our approach.

2.1 Illumination Estimation in Outdoor Environments

Because the main light source on outdoor environments is the Sun, the studies related to estimating the illumination on outdoor environments are mainly focus on finding the Sun position. In some studies, the skylight is considered as the ambient light of the scene in addition to the Sun. There are also some studies that assume that the Sun position is known and try to find the change in intensity

of the sunlight during a period of time.

Panagopoulos et al. [2] suggest a graphical model called as high-order Markov Random Field (MRF) illumination model. This model uses a 3D model of an object in the scene and the estimated shadow of the same object in order to estimate the illumination from a single image.

Lalonde et al. [3] use some cues that may exist in an image such as sky appearance, vertical surfaces, ground shadows and appearance of pedestrians. By combining the data extracted from each cue, they try to estimate the Sun position in a single outdoor image.

In another study that tries to estimate the Sun position from a single image, Liu et al. [4] first detect a known object on the image and find the surrounding shadows of that object. They then illuminate the 3D model of the known object from different angles and try to approximate to the shadows on the original image. The illumination position that gives the best approximation is accepted as the estimated sun position.

Lalonde and Matthews [5] try to estimate illumination from outdoor image collections. They first reconstruct a 3D model of a place by using its photographs taken by different angles. They then estimate the illumination on each photograph by training these photographs with their ground truth high dynamic range (HDR) lighting conditions and using the reconstructed 3D model.

Assuming that the Sun position is known, Liu and Granier [6] track the changes in intensity of the sun light in a video sequence. They analyze the extracted feature points on non-shadowed flat surfaces and according to the changes on these points, they try to estimate the intensity of the sun light in a time period.

Andersen et al. [7] estimate the dynamic light changes in an outdoor scene. In an offline preprocessing stage, they acquire diffuse reflectance properties by using inverse rendering techniques. These reflectance properties are then used to estimate the illumination properties in an online procedure. This method has some assumptions, such as known Sun position, simple 3D model with significant

surfaces, predefined diffused surfaces and HDRI environment map.

In a similar study, Xing et al. [8] try to estimate the dynamically changing illumination parameters of outdoor video sequences captured by a fixed camera. Their approach has an initialization stage that includes capturing an image in early morning and manually labeling upward diffuse surfaces in the image. By using the information acquired at the initialization stage, the skylight and sunlight parameters are estimated online.

2.2 Illumination Estimation in Indoor Environments

Illumination estimation in indoor environments may be challenging because these environments may have more than one dominant light sources and these light sources may have different shapes. Most of the studies related to indoor environments calculate the radiance on the scene instead of estimating the position of the light sources. For this purpose, they need the 3D model or geometric information of the scene and this is achieved mostly by using RGB-D cameras and depth maps.

Ikeda et al. [9] generate a radiance map on the scene by using incomplete object shape captured by a depth camera and the estimated shadow of the object. Similarly, Gruber et al. [10, 11] construct the 3D scene geometry in real-time by using an RGB-D camera and develop a radiance transfer method that defines the interactions between scene objects in order to calculate the global illumination on the scene. Lensing and Broll [12] also use the depth images from RGB-D camera and combine them with reflective shadow maps in order to estimate the global illumination on the scene in real-time. In another study, Neverova et al. [13] first separate the color image into specular and diffuse components. Then, by rendering these images in different forms with the help of a depth image, they approximate to the original image using an optimization process.

Yoo and Lee [14] estimate the light sources from HDR images. They first segment an HDR image into bright, medium bright and dim images. Then, they estimate the positions of the light sources from these segmented images by using the ratio of intensity of radiation. Lopez-Moreno et al. [15, 16] estimate the directions and intensities of the multiple light sources in an indoor image by using the silhouette of an object marked by the user and image processing techniques.

2.3 Illumination Estimation for Mobil Devices

The usage of AR applications on mobile devices such as smart phones and tablets increased recently. However, achieving the photo-realistic illumination estimation with these applications is limited due to the low computational power of such devices and dynamically changing user environment.

Rohmer et al. [17] suggest an interactive illumination estimation method on mobile devices. They try to handle the computational power problem by sharing the necessary computation between a stationary PC and mobile devices. They place multiple HDR video cameras in the environment that capture the entire scene visible to the mobile device in order to estimate the illumination.

In another study related to the mobile devices, Arief et al. [18] use a three-dimensional (3D) AR marker, which is also a reference object in the scene, and analyse the relationship between this reference object and its shadow in order to estimate the position of a single light source in the environment.

2.4 Other Studies for Illumination Estimation

Because the illumination estimation is a challenging problem, especially in AR environments, there are many studies that try to solve this problem by using different methods and try to obtain good results. The methods other than the

ones mentioned in the above sections generally use reference objects, different camera technologies or they aim to solve specific cases.

In the studies by Debevec [19], Kanbara and Yokoya [20], Wong et al. [21], and Lee and Jung [22], a mirror ball placed on the scene is used as a light probe. In these studies, the bright points on the mirror ball are evaluated together with the position of the ball according the camera and the position of the light sources on the environment are estimated. In a similar way, Nishino and Kayar [23] use the human eye as a natural light probe. By using the shape of the cornea and the camera viewing it, they construct a spherical environment map of the scene and from this environment map they estimate the illumination of the scene. Sato et al. [24] and Yoo and Lee [25] both use fisheye lens cameras to obtain the omnidirectional images of the scene and then they use these images in order to calculate the direction of the light sources.

Marschner and Greenberg [26] estimate the directional distribution of the incident light by using a photograph and a 3D model of the pictured object. With the help of the camera, they first generate a set of basis images by using the 3D model and a set of basis lights. They then use a linear system inversion method to find a linear combination of these basis images that matches the original photograph. The coefficients in this linear combination give the lighting solution.

Sato et al. [27] obtain the illumination distribution of a scene from a radiance distribution inside shadows cast by an object of known shape onto another object surface of known shape and reflectance. They estimate the illumination distribution of the real scene as a set of imaginary point light source distributions over the scene by using the radiance distribution inside the shadows.

In another study, Knorr and Kurz [28] use human face for illumination estimation. By analyzing some specific points on a human face and learning radiance transfer functions of these points from a data set consisting of face images with known illumination, they estimate the lighting conditions in real time.

2.5 Illumination Estimation for Time-lapse Videos

There are not so many studies that relate the illumination estimation with time-lapse videos or track the sun position change during the time-lapse video sequences. However, an increase in the number of studies in this area is expected in the near future because of the growing popularity of time-lapse videos.

Sunkavalli et al. [29] and Zhang et al. [30] both decompose each image in the time-lapse video into sunlight and skylight basis images by using the information from whole video sequence. The aim of these studies is to relight the scene easily, to recover a portion of the scene geometry and to perform some image editing operations. For this purpose, Sunkavalli et al. analyse the points in shadow and direct sunlight by using matrix factorization. As a result of this, they obtain the per-pixel offsets, the basis curves that describe the intensity changes over time, and the scales of these basis curves that are useful to obtain the spatial variation of reflectance and geometry. Using this information, they estimate the basis images illuminated only by the Sun and only by skylight. Zhang et al. also follow a similar approach. They first detect the shadowed regions by evaluating the value of each pixel along the video sequence with k-means clustering. With this information, they obtain the skylight basis images. They then estimate the sunlight basis images by calculating the basis curves.

Lalonde et al. [31] transfer appearance and illuminant from time-lapse sequences to other time-lapse sequences or single images. To this end, they first build a Webcam database that consists of lots of time-lapse sequences. In order to transfer appearance to an original image, they evaluate illumination conditions which are the sun position, sky color and weather conditions of the original image. They find an image from the Webcam database with similar illumination conditions and transfer an object from that image to the original image. To achieve illuminant transfer, they propose a model to obtain the high dynamic range environment maps of the images. They use the acquired environment maps to illuminate the virtual objects into the images.

2.6 Illumination Estimation that Works for Both Indoor and Outdoor Environments

We need to estimate the initial position of the Sun before tracking the shadows to calculate the change in the position of the Sun during a time-lapse video sequence. For this purpose, we use the approach of Chen et al. [32], which tries to estimate the scene illumination from a single image, and modify it according to our needs. The main advantage of this approach is that it tries to estimate the illumination in both outdoor and indoor environments. Because we are dealing with environments where the Sun is the main light source, these may include both outdoor and indoor environments whose main light source is the Sun like shops or cafeterias, as shown on Figure 2.1. We estimate the initial position of the Sun from the first frame of the time-lapse video sequence using the approach proposed by Chen et al.

Figure 2.1: Sample outdoor environment (left), sample indoor environment with the Sun as the main light source (right)

The approach proposed by Chen et al. consists of three stages. They try to construct the scene geometry from a single image by using some existing methods. They then decompose the image into its intrinsic components, which are reflectance and shading image. Finally, as a result of an optimization process that uses information from the geometry and the shading image, they estimate the environment illumination. Their approach do not estimate the exact positions of the light sources; instead, they approximate the environment illumination with the light sources placed symmetrically on a hemisphere covering the scene. In the sequel, we describe how we modify their approach for time-lapse videos.

Chapter 3

Estimation of the Initial Position of the Sun

In order to keep track of the position of the Sun during a time-lapse video, we first need to estimate the initial position of the Sun. Although there are a lot of studies that estimate the Sun position accurately, these studies are generally for the outdoor environments. However, the approach proposed by Chen et al. estimates the illumination in both outdoor and indoor environments and it is more suitable to our needs because we are dealing with the position of the Sun in both outdoor and indoor environments where the main light source is the Sun. Because Chen et al. mainly aim to estimate the position of the Sun in a single image, we suggest some improvements for different stages of their approach to make it applicable to time-lapse videos.

3.1 Geometry Extraction

Chen et al. require the geometric model of the scene represented by the image. This is important to obtain the normal vectors of surfaces that exist in the scene,

which are used for illumination calculations. For this purpose, they use the approach proposed by Saxena et al. [33], which aims to estimate the scene geometry from a single image. Saxena et al. use some image features, such as color, texture, edge and location, together with the Markov Random Field (MRF) model. They get qualitatively correct results in approximately 65% of the test images by accepting an image as correct when the 70% of the major planes on that image is correct. We observed that this accuracy is not sufficient to estimate a correct position of the Sun because the scene geometry extracted using their approach includes lots of incorrect surface normals. Chen et al. also point out that the estimated geometry is not quite accurate in the discussion part of their paper.

Instead of extracting the scene geometry from the image, we prefer to obtain the coarse 3D model of the scene manually. Even though manually modeling the 3D scene geometry requires a preprocessing stage, it is important to obtain more accurate results. In a study whose only aim is to estimate the position of the Sun in a single image, the geometry extraction from the image generally give acceptable results. However, for estimating the initial position of the Sun for time-lapse videos, the inaccurate results in the initial stage will affect the later stages severely.

3.2   Intrinsic Components

On the second stage of the illumination estimation, Chen et al. decompose the input image into its intrinsic components, which are reflectance and shading images (Figure 3.1 (b), (c)). An image can be considered as per-pixel product of its intrinsic reflectance and shading components. The shading image obtained as a result of the decomposition can be accepted as the irradiance of the surface and used in the illumination model.

(a) Input Image
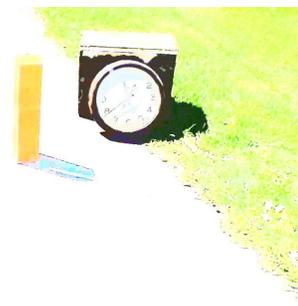
(b) Reflectance Image

(c) Shading Image

(d) Grayscale Image

Figure 3.1: Intrinsic components of an image. Starting from the input image (a), we compute the reflectance image (b), and the shading image (c). The grayscale image (d) is brighter, but it is very similar to (c).

Although there are many studies that decompose the images into their intrinsic components, Chen et al. indicate that one of the approaches proposed by Tappen et al. [34], Shen et al. [35], or Jiang et al. [36] can be used for this purpose. We decided to use the approach proposed by Garces et al. [37]. We observed that the shading image obtained using their decomposition method is similar to a simple grayscale image converted from the original RGB image (cf. Figure 3.1 (d)). We observed that even though the intensity values of the grayscale image are approximately five to seven units more than the intensity values of the shading image, this difference have a negligible effect on the results. Using the grayscale image instead of the shading image does not change the estimated position of the Sun and causes only a slight difference on light intensity of the Sun. We prefer using the grayscale image because of its small computational cost.

3.3   Illumination Model

In this stage, our aim is to estimate the approximate position of the Sun from the first frame of the time-lapse video by using the 3D model of the scene and the grayscale image that we acquire on the previous stage. For this purpose, we use the illumination estimation method proposed by Chen et al. tailored to our needs.

We first place eight spotlights over the 3D model of the scene homogeneously and symmetrically in such a way that these spotlights will construct a hemisphere over the scene. Each spotlight is directed to the center of the constructed hemisphere and their lighting ranges are adjusted in such a way that they enclose the 3D model entirely. These spotlights approximate the real world illumination caused by the main light sources on the scene. Apart from these spotlights, a point-light is also placed over the hemisphere in order to obtain the ambient light coming from the other points on the scene other than the main light sources. Figure 3.2 depicts a sample model with its light sources.

In the second step, we find the corresponding positions and normal values of
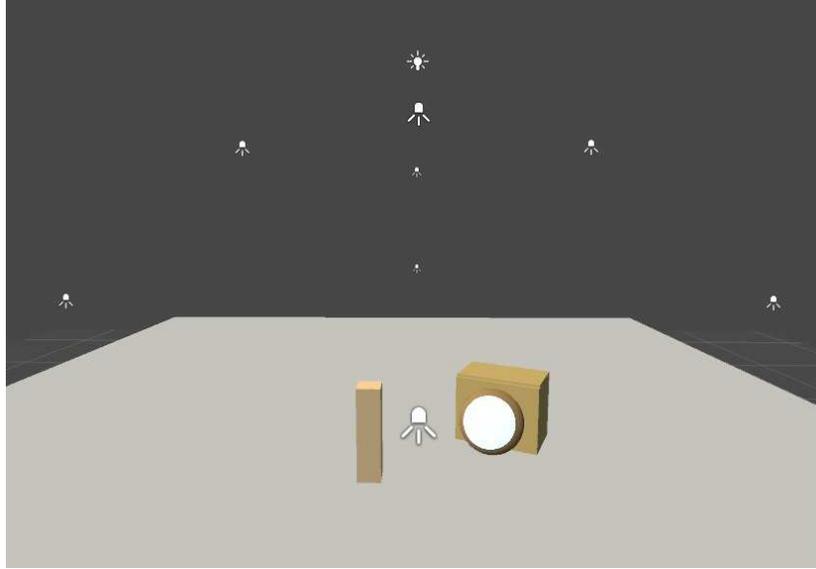
Figure 3.2: A 3D model with eight spotlights and one point light.

the pixels on the 3D model. For this purpose, the image is matched with the 3D model of the scene by using the camera tracking algorithm developed by [38] (cf. Figure 3.3). After this matching step, a ray is cast from the camera position to each pixel in the image. The first point where the ray cast intersects the 3D model is the geometric position of the pixel. The normal value of the pixel can also be found easily.

In the next step, we determine the spotlights that contribute to the illumination of each pixel. For this purpose, we cast a ray from each of eight spotlights to the geometric position of a pixel. If the first point where the ray cast from a spotlight intersects the 3D model is equal to the geometric position of the pixel to which the ray is cast, then the spotlight contributes to the illumination of the pixel. Otherwise, the pixel is blocked by some parts of the model and that spotlight does not contribute to the illumination of the pixel. From this point, if we subtract the position of a spotlight from the position of a pixel, we can also obtain the light vector from spotlight to the pixel, which we use for illumination calculation.

On Lambertian surfaces, the irradiance can be represented by [39]:
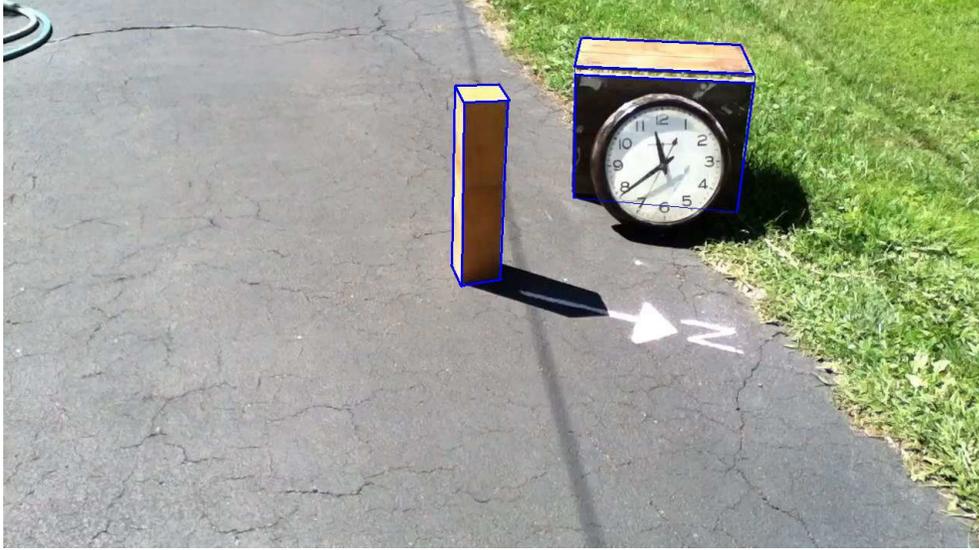
Figure 3.3: Image matched with the 3D model using the tracking algorithm.

$$S = I_a + \sum_{i=1}^{m} I_i\, L_i \,\cdot\, N, \qquad (3.1)$$

where S is the pixel values of the grayscale image, I_a is the ambient light, I_i is the intensity of the ith light source reaching to the pixel positions, L_i is the direction of the ith light source to the pixel positions, m is the number of the light sources that illuminate the pixels, and N is the normal values of the pixel positions. Among these variables, we know S, L_i and N and we need to find I_a and I_i. If we apply the Levenberg-Marquardt minimization algorithm [40] between the grayscale image and the estimated irradiance, we can obtain the intensity values of the eight spotlight sources and the point-light source required to approximate to the real world illumination:

$$\arg\max_{(I_a, I_1, I_2, \dots, I_8)} \sum_{j=1}^{n_s} \left(S_j - \left(I_a + \sum_{i=1}^{8} I_i\, L_i\, . \, N \right) \right), \qquad (3.2)$$

where S_j is the value of the jth pixel on the grayscale image and n_s is the total number of pixels.

16

Although we obtain the intensity values of the spotlights and point-light we placed over the 3D model, in the environments where the main light source is the Sun, the source of illumination is mostly concentrated on one or two spotlights, as it is expected. For the next stage of our framework, we need to find a single position for the Sun so that we can easily track its movement. We determine the center of intensity formed by the spotlights on the hemisphere as the estimated position of the Sun and place a directional light source to that position instead of the eight spotlights. The point light source that we place over the hemisphere is used to account for the ambient light. The overview of the algorithm is given in Figure 3.4.
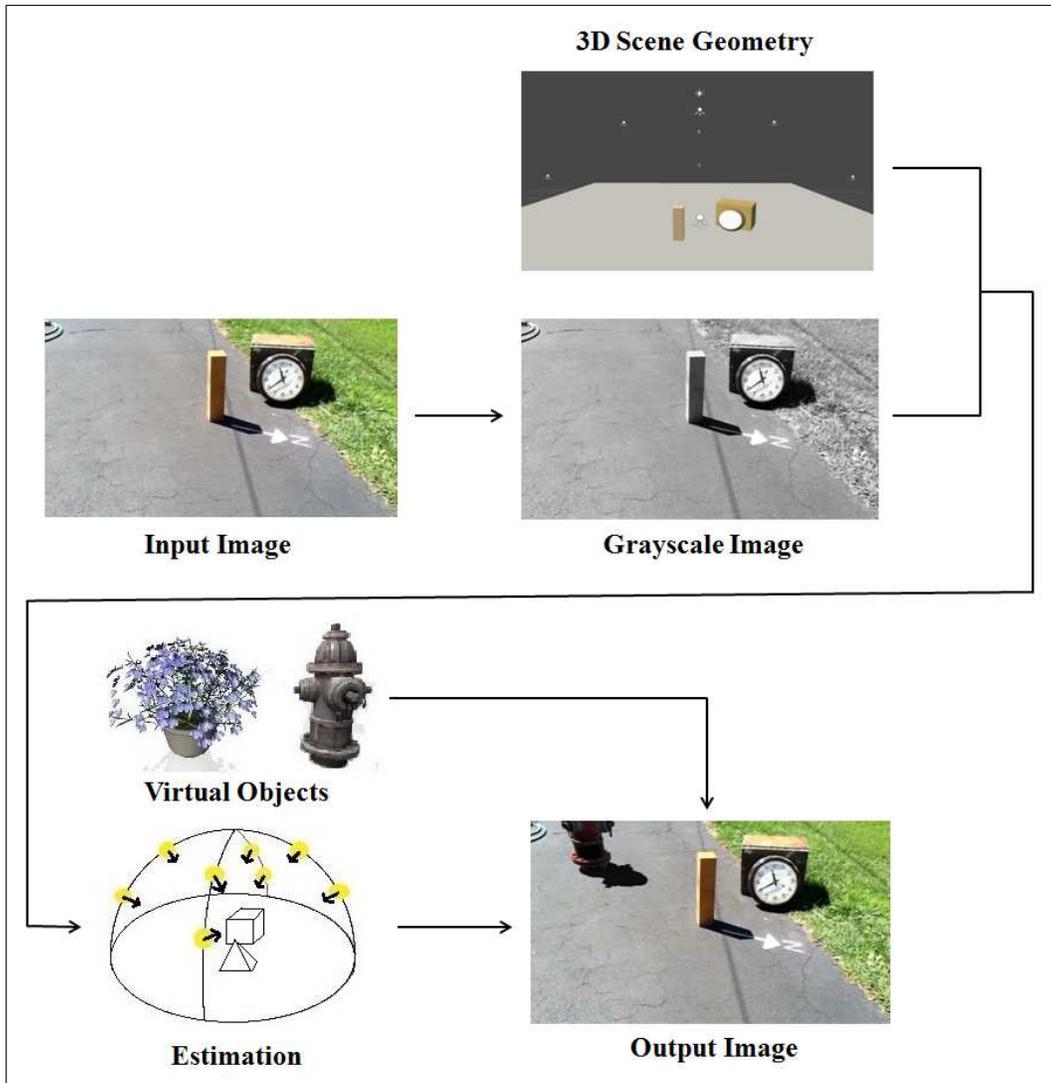
Figure 3.4: The overview of the algorithm that estimates the initial position of the Sun.

Chapter 4

Shadow Tracking-based Estimation of the Position of the Sun

After the estimation of the initial position of the Sun from the first frame of the time-lapse video, we estimate the position of the Sun in the following frames. Because the method that we use on the first frame is costly, we cannot apply it on every frame. To estimate the Sun position fast and accurately for the remaining frames of the video, we propose a method that uses hard ground shadows on the Sun direction, which we assume that there exists at least one hard shadow in the time-lapse video. This method tries to estimate the position of the Sun at each frame by calculating the changes in the length and direction of these hard shadows.

First, the hard ground shadows on the first frame are determined by using a state-of-the-art method [41]. Then, these shadows are eliminated according to a set of specific criteria (described in detail below) and the energy image of the first frame. As a result of the elimination, the most appropriate shadow for tracking is determined. On the following frames of the time-lapse video, this shadow is tracked by using a pixel-wise method that uses the energy images of these frames. According to the changes in the length of the shadow and its direction, the zenith and azimuth angles of the Sun are estimated and the position of the

Sun is updated accordingly.

4.1 Shadow Selection

In order to estimate the Sun position during a time-lapse video, we use hard ground shadows. We think that the ground shadows are more informative in estimation of the zenith and azimuth angles of the Sun whereas the shadows on the other surfaces may lead to misleading information. The reason we use the hard shadows is that the hard shadows are easier to track then the soft shadows and the soft shadows may be unstable during the video.

Although the hard shadows may be good at the estimation of the Sun position, every hard ground shadow may not work for our purpose. Therefore, we try to determine a single, appropriate hard ground shadow to track during the time-lapse video. For this purpose, firstly we determine the hard ground shadows on the first frame of the time-lapse video by using the method proposed by Lalonde et al. [41]. This method tries to detect the ground shadows in a single image and produces really good results. An example output of this method can be seen in Figure 4.1. In addition to finding the hard ground shadows, we also need to extract the energy image of the first frame that we use in the elimination process of the hard ground shadows. An energy image can be generated by applying an energy function to each pixel in an image. We use the energy function proposed by Avidan and Shamir [42]:

$$e(I) = \left| \frac{\partial}{\partial x} I \right| + \left| \frac{\partial}{\partial y} I \right|, \tag{4.1}$$

where I is the input image. We apply this energy function to the first frame of the time-lapse video and obtain its energy image. Figure 4.1 shows an example energy image. We use the energy images during the shadow tracking process in the subsequent frames.
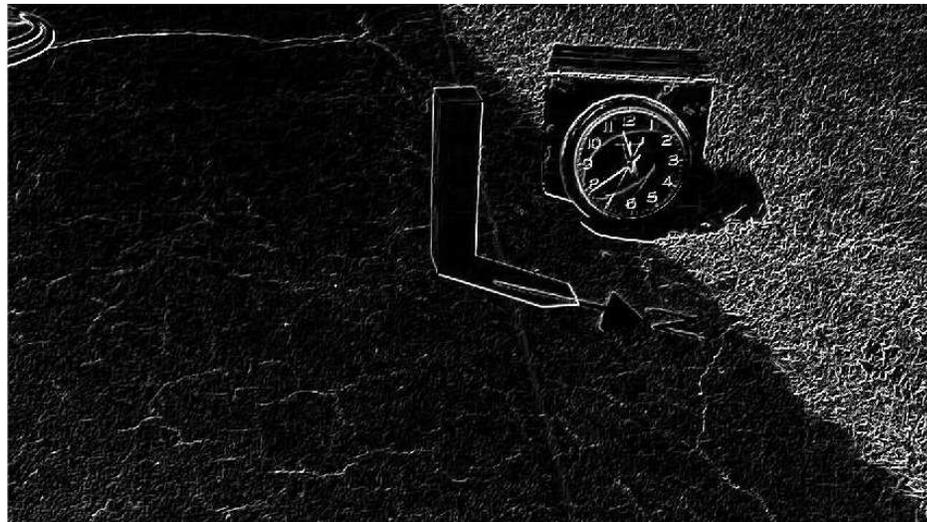
Figure 4.1: An output image with shadow edges shown in blue (top), and a sample energy image (bottom).

We eliminate all but one of the hard ground shadows and decide the most suitable one to be used in the rest of the frames according to the following five steps in the given order:

1. If the angle between a shadow and the estimated initial position of the Sun is greater than 10°, then that shadow is eliminated. These shadows do not give accurate information about the changes on the position of the Sun. Because the initial position of the Sun is estimated with at most 10° error, we do not want to miss a shadow that is really on the direction of the Sun.

2. If at least one end of a shadow is on the boundary of the frame, that shadow is also eliminated. This is because the whole shadow is not seen in the frame. In other words, the visible part of the shadow may disappear or the invisible part may appear in the following frames and this may cause errors in the results.

3. The shadows whose energy values are lower than the average energy value are eliminated. Firstly, a shadow energy value is calculated for each shadow by taking the average of the energy values of the pixels that form the shadow. The energy values of the pixels are taken from the energy image generated previously. Then the average of the all shadow energy values is calculated and the shadows whose shadow energy values are lower than this average value are eliminated. We try to keep the shadows with high energy value because their tracking is easier than the ones with low energy value.

4. The average shadow length is calculated from the remaining shadows and if a shadow is shorter than the average length, that shadow is eliminated. This is because it is hard to obtain significant information about the changes in the length and direction of short shadows and long shadows provide more precise information in this sense.

5. From the remaining shadows, if there exists shadows whose only one corner intersects with corner of an object in the 3D model, we choose the shadow with the highest energy from such kind of shadows as the final shadow. If

there is not such a shadow, we search for the shadows whose both corners do not intersect any object in the 3D model and select the one with the highest shadow energy. If again there is not such a shadow, we choose the shadow with the highest energy as the final shadow. To decide whether or not the corner of a shadow intersects with the corner of an object on the model, we use the matched image-model pair described previously.

4.2   Shadow Tracking Algorithm

4.2.1   Shadow Tracking Algorithm

We need to define some variables that will be used in the proposed shadow tracking algorithm. We define the pixel that is closest to the estimated position of the Sun from the shadow pixels as the start pixel (s). Related with this, s_x and s_y denote the horizontal and vertical pixel distances of the start pixel s to the top-left corner of the frame, respectively. Similarly, we define the pixel on the other corner of the shadow as the final pixel f. f_x and f_y denote the horizontal and vertical pixel distances to the top-left corner of the frame, respectively. We define the shadow length l_s as follows:

$$l_s = \sqrt{(f_x - s_x)^2 + (f_y - s_y)^2}. \tag{4.2}$$

Let z and a denote the zenith and azimuth angles of the Sun, respectively. z and a are initially equal to the zenith and azimuth angles of the Sun estimated for the first frame and will be updated in subsequent frames. The length of the object that generates the shadow we are tracking, l_v, is used to calculate the zenith angle of the Sun in subsequent frames and it can be calculated according to the following equation:

$$l_v = l_s \tan z. \tag{4.3}$$

Let θ denote the counterclockwise angle that the shadow edge makes with the positive x-axis. It can be calculated according to the following equation:

$$\theta = \begin{cases} 360 - arccos\left(\dfrac{\vec{t}\cdot\vec{v}}{\sqrt{(f_x-s_x)^2+(f_y-s_y)^2}}\right), & \text{if } s_y < f_y \\ arccos\left(\dfrac{\vec{t}\cdot\vec{v}}{\sqrt{(f_x-s_x)^2+(f_y-s_y)^2}}\right), & \text{otherwise} \end{cases} \tag{4.4}$$

where $\vec{t}$ is a vector from $(s_x,\ s_y)$ to $(s_x+1,\ s_y)$ and $\vec{v}$ is a vector from $(s_x,\ s_y)$ to $(f_x,\ f_y)$.

After the preprocessing on the first frame of the time-lapse video, we track the shadow and estimate the position of the Sun in the following frames. This process is described in Algorithm 1.

Algorithm 1 Shadow tracking and estimation of zenith and azimuth angles of the Sun

---

INPUT: time-lapse video, start pixel s, azimuth angle a, object length l_v
OUTPUT: updated position of the Sun

1: **while** not the end of the video **do**
2:     fetch the frame into a matrix
3:     extract the energy image of the frame
4:     determine the search direction according to a
5:     update s using s of the previous frame
6:     *check* ← true
7:     **while** *check* **do**
8:         search the next pixel p_n, which constructs the shadow according to the search direction determined using a
9:         **if** energy value of p_n < energy threshold **or** slope of shadow > slope threshold **then**
10:             *check* ← false
11:             $f \leftarrow p_n$
12:         **end if**
13:     **end while**
14:     calculate z using f, s and l_v
15:     calculate a using f, s of the current frame and the a value of the previous frame
16:     update Sun position according to new z and a values
17: **end while**

First, we fetch the next frame on the video sequence and extract its energy

image. We check whether the start pixel s that comes from the previous frame has still the same energy value in the current frame. If its energy value does not change, we continue to use s as the start pixel. Otherwise, we search for a new start pixel to construct the shadow on the current frame. We determine the new position of the start pixel by searching the pixel that is the center of mass of a 5×5 pixel area in terms of energy values, where the center pixel is the old s (cf. Figure 4.2). $e_{i,j}$ denotes the energy value of the pixel on the ith row and the jth column inside the pixel area. We can find the center of mass of the pixel area by using the following equations:

$$m_i = round\left(\sum_{i=1}^{5}\sum_{j=1}^{5} j\, e_{i,j}\right), \tag{4.5}$$

$$m_j = round\left(\sum_{j=1}^{5}\sum_{i=1}^{5} i\, e_{i,j}\right), \tag{4.6}$$

where m_i and m_j denote the row and column indices of the center of mass within the 5×5 pixel area. We assign the center of mass as the new start pixel by using the following equation:

$$s = (s_x - 3 + m_j,\, s_y - 3 + m_i). \tag{4.7}$$

Before we start the construction of the shadow in the current frame, we need to determine the search direction for possible shadow pixels. For this purpose, we use the azimuth angle a calculated for the previous frame and we choose one of the search directions, as shown in Figure 4.3.

Starting from s, we move in the chosen search direction. While selecting the next shadow pixel p_n, we choose the pixel with the maximum energy from four possible pixels on the search direction. We continue in this fashion by constructing the shadow that we track to estimate the zenith and azimuth angles of the Sun until one of two termination conditions occurs:
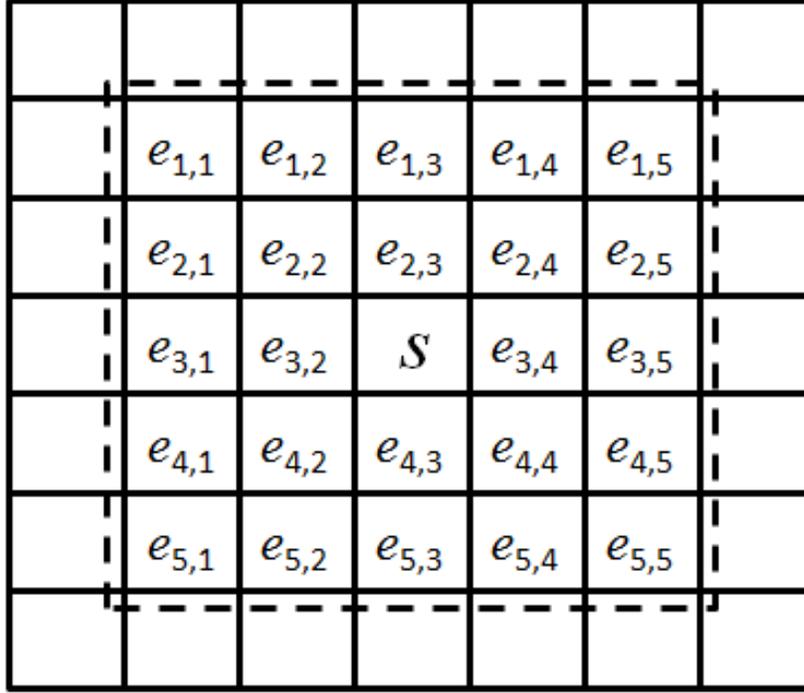
Figure 4.2: Searching the new start pixel. 5×5 pixel area with the old start pixel s in its center is the search area for the new start pixel. The pixel that is the center of mass in terms of energy values will be the new start pixel.

1. We stop finding the next pixel if the energy value of a pixel is less than the energy threshold. We define the energy threshold as the half of the energy of the initial shadow in the first frame. A pixel whose energy is below the threshold means that we reach the other corner of the shadow. This condition is illustrated in Figure 4.4. If the energy value of the shadow is 230, the energy threshold is 115. Assume that the azimuth angle a from the previous frame is 43°. We search the pixels starting from s on the appropriate search direction. When we reach the pixel with the energy value 100, we stop searching and accept the pixel as the final pixel f because its energy value is less than 115.

2. While constructing the shadow in the current frame, we calculate the slope of the shadow after the 15th pixel. Then, at every 10 pixels, we calculate the slope of the last 10 pixels and if the last calculated slope differs at least one third of the slope of the first 15 pixels, then we stop the shadow construction process. Because it means that we reach to the other corner of the shadow

26
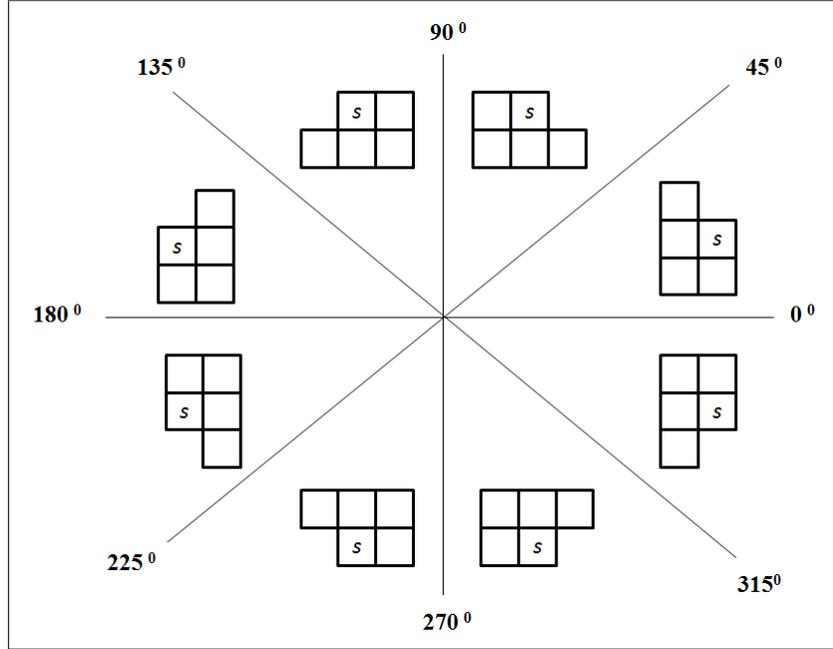
Figure 4.3: Determining the search direction. According to the value of the azimuth angle a, the appropriate search direction is chosen to form the shadow in the current frame. A pixel with label s denote the start pixel and the other four pixels are the candidates for shadow pixels.

and probably a new shadow in another direction starts. An example of this condition can be seen in Figure 4.5. In this figure, the yellow pixel shows the 15th pixel of the shadow. As it is seen, because the slope of the last 10 pixels after the green pixel differs from the slope of the first 15 pixels at least by its one third, we stop searching new pixels and accept the green pixel as the final pixel f.

We determine the energy and slope thresholds of the termination conditions as a result of the repeated experiments. In our experiments, we observe that the second termination condition is encountered more frequently than the first one.

After we find the starting and ending pixels of the shadow in the current frame, we calculate the new values of the zenith and azimuth angles. We calculate the zenith angle, z, and the azimuth angle, a, according to the following equations:

							225		
						225	230	s	
					215	225	(235)	230	
				200	220	(240)	235		
			190	215	(230)	225			
		80	200	(225)	220				
	50	95	(215)	210					
	75	(100)	80						
	80	70							

Figure 4.4: The termination condition that depends on the energy threshold.

Figure 4.5: The termination condition that depends on the slope threshold.

$$z = \arctan(l_v/l_s), \tag{4.8}$$

$$a = a_{prev} + (\theta_{curr} - \theta_{prev}), \tag{4.9}$$

where a_{prev} denote the azimuth angle for the previous frame, θ_{curr} and θ_{prev} denote the values of the angle θ (cf. Equation 4.4) for the current and the previous frames, respectively. The position of the Sun can then be updated using the new zenith and azimuth angles.

Chapter 5

Evaluation and Results

In studies related to the illumination estimation, the results are generally evaluated qualitatively with visual outputs and quantitatively by comparing the resulting angles/directions of the light sources/shadows with ground truth values. We present some result images from the time-lapse videos with some virtual objects placed seamlessly in order to show the qualitative success of our study. We evaluate our study quantitatively by comparing the estimated zenith and azimuth angles of the Sun with ground truth values.

The time-lapse videos used for testing this study are taken from video-sharing website Youtube. We used a notebook computer with Intel i7-4700MQ (2.4GHz Clock) processor, 6Gb RAM, AMD Radeon HD 8750M GPU to make the required computations and tests in this study. The parts related to the initial estimation of the position of the Sun are implemented mostly on the Unity game engine [43]. The shadow-tracking-based Sun position estimation is implemented mostly in MATLAB [44].

It is possible to apply the proposed method either online or in two-passes. In the online case, we determine the azimuth and zenith angles for each frame and apply a smoothing procedure by using exponential smoothing and linear regression to these azimuth and zenith values to find the Sun position and illuminate

virtual objects accordingly. In the two-pass case, we first find the azimuth and zenith angles for each frame and approximate these values using linear curves at the end of the first pass. In the second pass, we use these smoothed azimuth and zenith values to determine the Sun position and illuminate virtual objects accordingly. In terms of the quality of the results, the results of the two-pass procedure are superior to those of the online procedure.

We present the results of the application of the method in two passes on three time-lapse videos [45, 46, 47]. The visual results on these videos can be seen by analyzing the illumination of the virtual objects placed into the videos in Figures 5.1, 5.3 and 5.5. Six still frames of each video show that the virtual objects are seamlessly integrated into the real scenes. The quantitative results that compare the estimated zenith and azimuth angles in these videos with the ground truth values are shown in Figures 5.2, 5.4 and 5.6. The initial differences between our results and the ground truth values show the error rates that are caused from the first stage of the algorithm that estimates the initial position of the Sun. Starting from these error rates, our shadow tracking approach works well in estimation of the position of the Sun, as it is seen from the parallel patterns between our results and the ground truth values.

Table 5.1 shows the error rates during the second stage of the proposed approach by giving the minimum, maximum and mean errors on three videos. The mean error rates on both zenith and azimuth angles are less than 6°. Because of the error in the initial Sun position, the length of the object that generates the shadow cannot be calculated exactly and this causes small increase/decrease on the error rates on the zenith angle. If the selected hard ground shadow is not in exactly the same direction with the Sun, this causes small changes on the error rates on the azimuth angle. According to our observations, the scenes with more light-shadow content affect the estimation of the Sun position positively. These scenes also include more alternatives to choose the appropriate shadow used in tracking process that affects the performance of the tracking stage of the approach positively.

	Scene 1	Scene 2	Scene 3
Zenith Min Error	4.38°	2.01°	1.04°
Zenith Max Error	6.54°	2.55°	4.17°
Zenith Mean Error	**5.46°**	**2.28°**	**2.60°**
Azimuth Min Error	3.7°	3.73°	3.05°
Azimuth Max Error	5.02°	5.81°	3.16°
Azimuth Mean Error	**4.38°**	**4.77°**	**3.10°**

Table 5.1: Error rates of the zenith and azimuth angles in the three videos

(a) 1st frame

(b) 85th frame

(c) 170th frame

(d) 255th frame

(e) 340th frame

(f) 420th frame
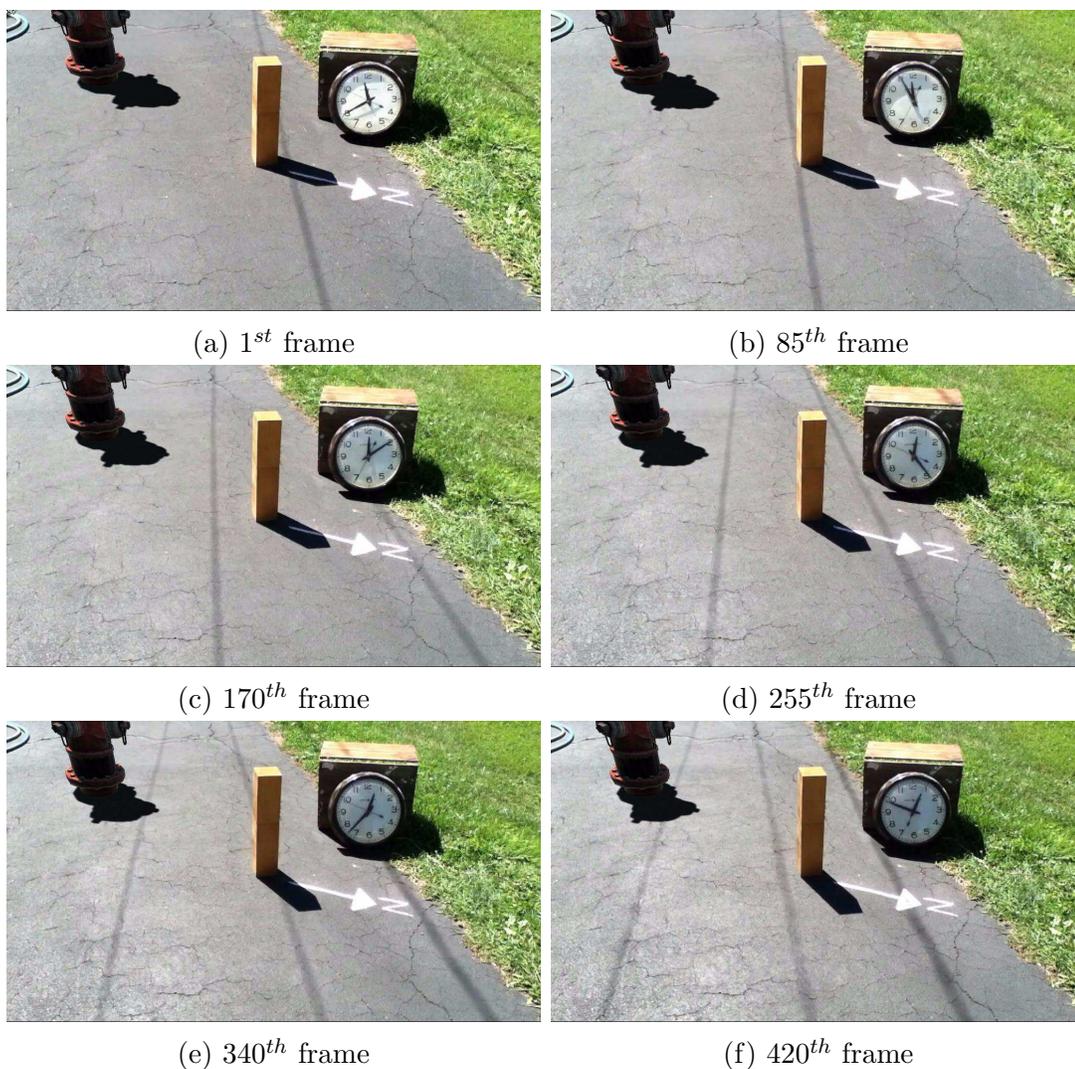
Figure 5.1: Still frames from our first time-lapse video with the position of the Sun changing. The illumination of the virtual fire hydrant changes synchronously with the position of the Sun.
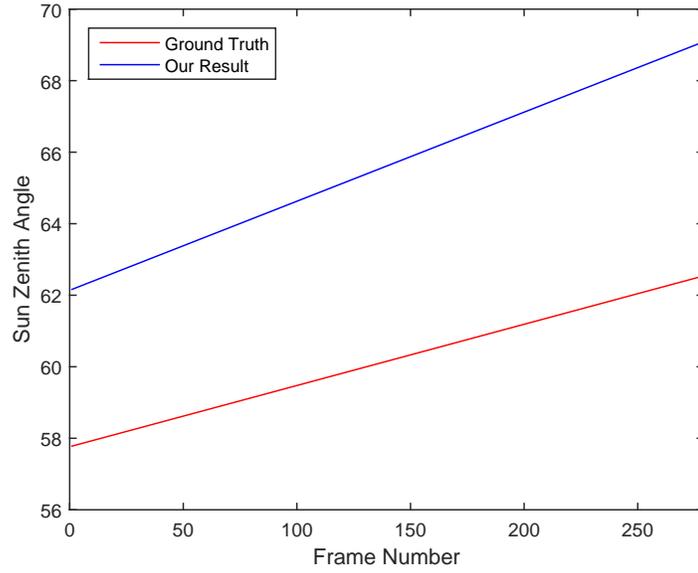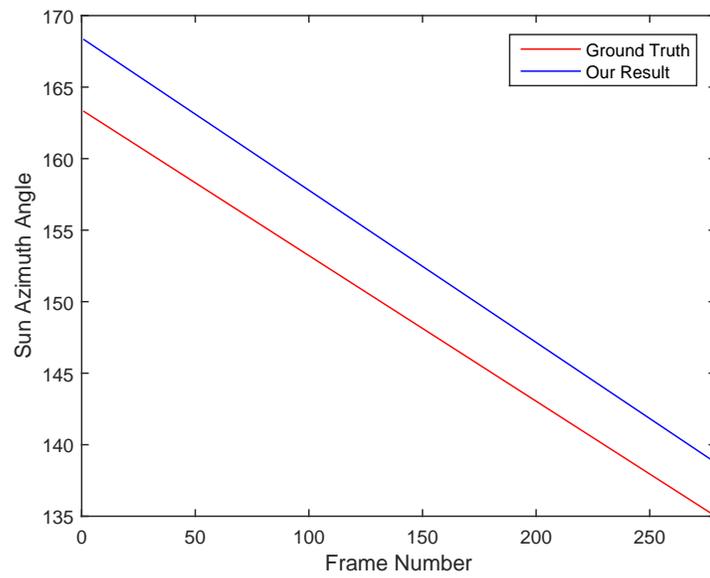
(a) Zenith Angles

(b) Azimuth Angles

Figure 5.2: Comparison of the zenith angles (a) and azimuth angles (b) of the first video with the ground truth values.

(a) 1st frame

(b) 125th frame

(c) 250th frame

(d) 375th frame

(e) 500th frame

(f) 620th frame
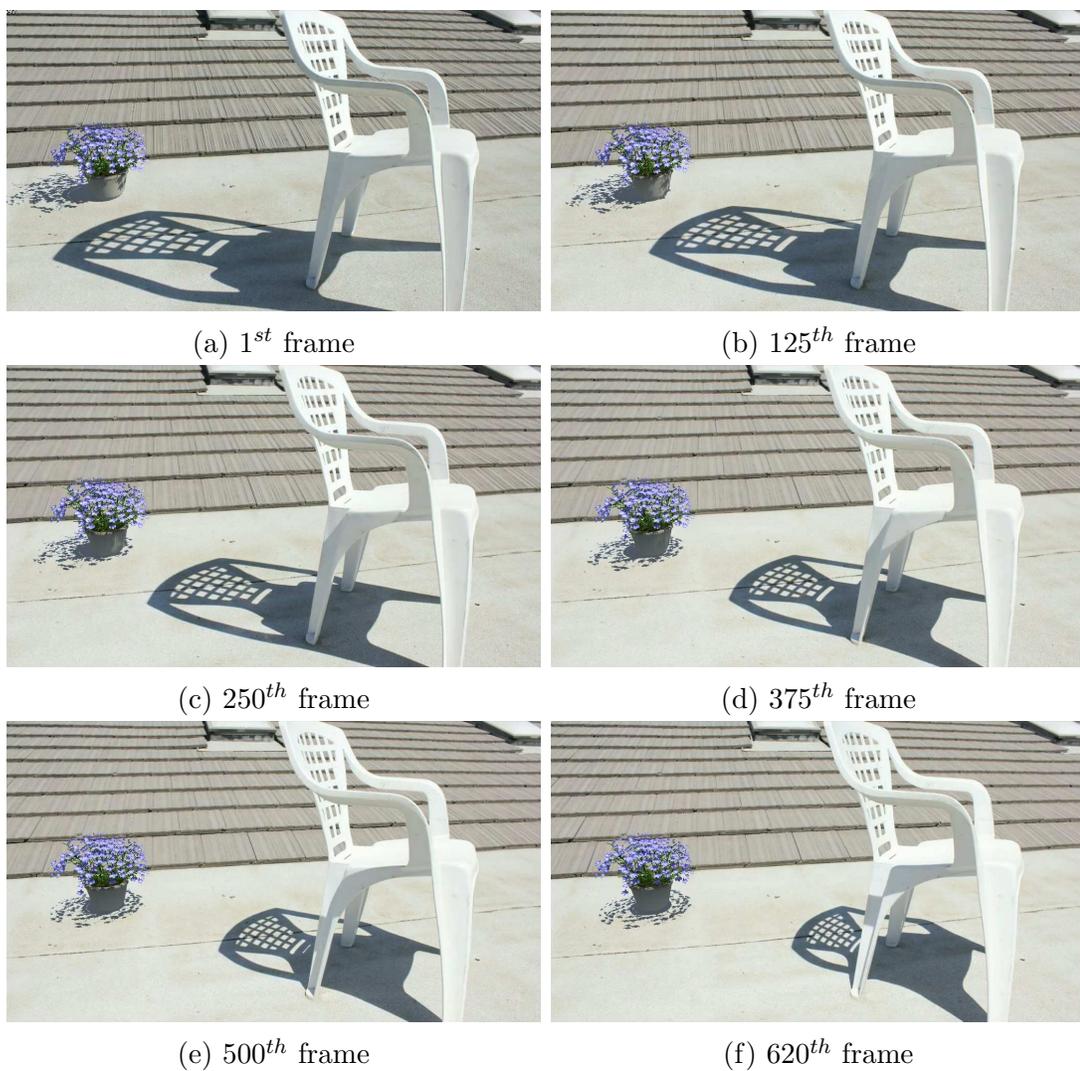
Figure 5.3: Still frames from our second time-lapse video with the position of the Sun changing. The illumination of the virtual flower changes synchronously with the position of the Sun.
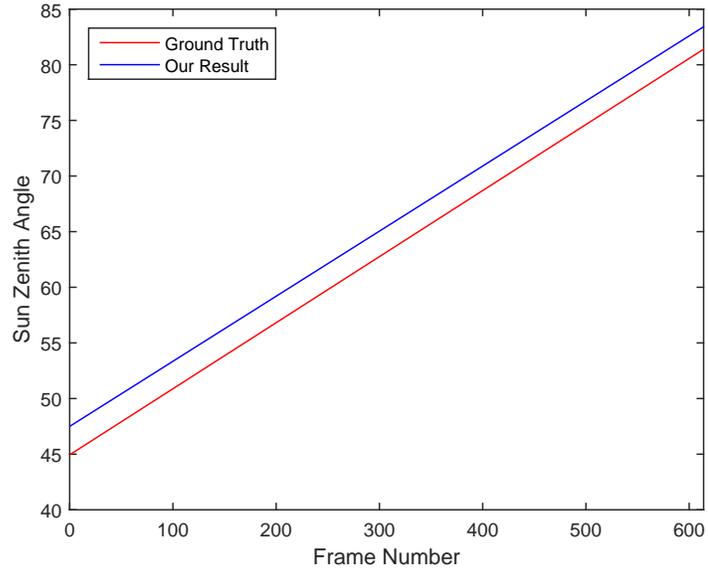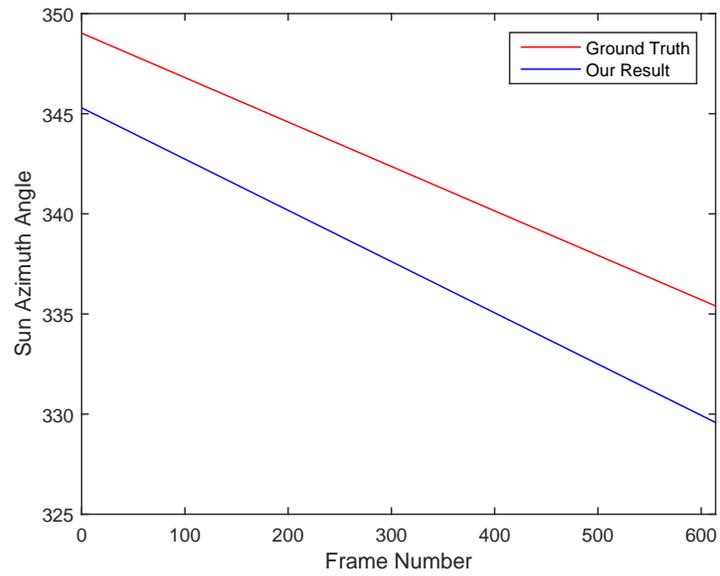
(a) Zenith Angles

(b) Azimuth Angles

Figure 5.4: Comparison of the zenith angles (a) and azimuth angles (b) of the second video with the ground truth values.

(a) 1st frame

(b) 40th frame

(c) 80th frame

(d) 120th frame

(e) 160th frame

(f) 200th frame

Figure 5.5: Still frames from our third time-lapse video with the position of the Sun changing. The illumination of the virtual trash bin changes synchronously with the position of the Sun.
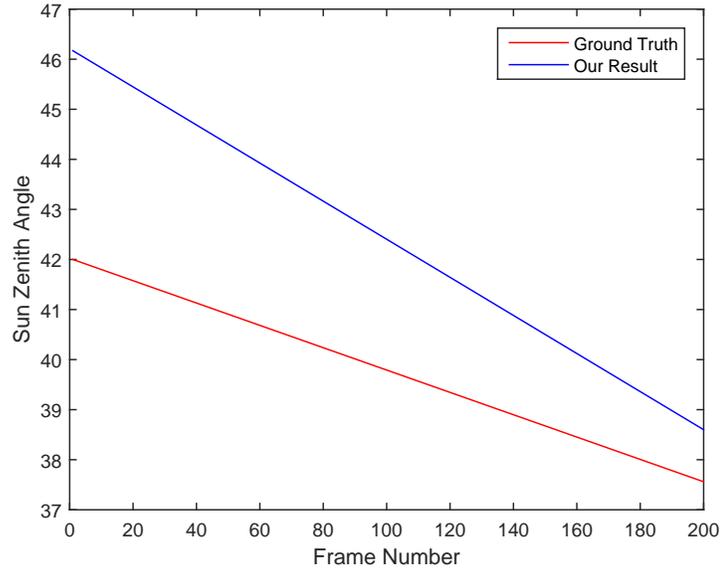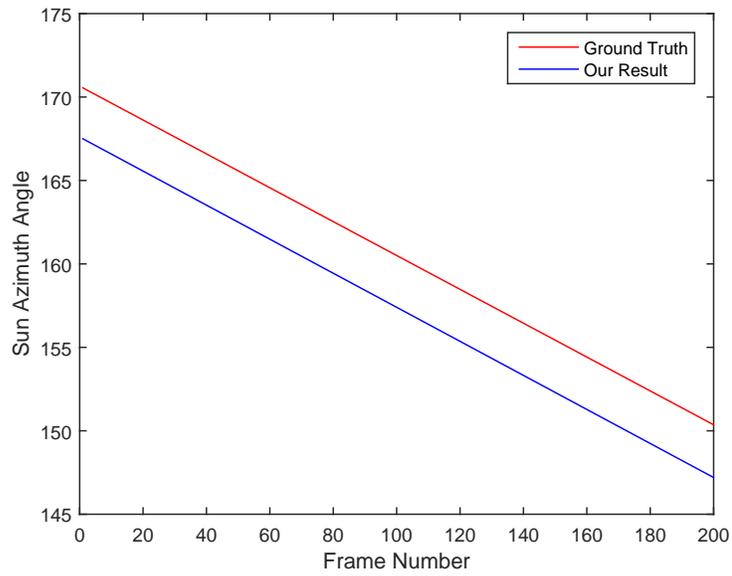
(a) Zenith Angles

(b) Azimuth Angles

Figure 5.6: Comparison of the zenith angles (a) and azimuth angles (b) of the third video with the ground truth values.

Chapter 6

Conclusions and Future Work

This thesis proposes a new method that facilitates the usage of time-lapse videos in Augmented Reality applications. To place virtual objects in real videos in a seamlessly illuminated fashion, we need to estimate the position of the light sources in real videos. We mainly target the videos of indoor and outdoor environments where the only light source is the Sun. We estimate the position of the Sun during a time-lapse video and calculate the illumination of the virtual objects placed in the real video accordingly. Our approach first estimates the initial position of the Sun from the first frame of the video by modifying an existing illumination estimation method. Then by tracking a hard ground shadow in the scene with an energy-based pixel-wise method for the rest of the video frames, it estimates the changes in the position of the Sun. The proposed method gives successful results on time-lapse videos found on the Internet.

The study has some drawbacks such as requiring a coarse 3D model of the environment prepared manually in a preprocessing stage for the estimation of the initial position of the Sun. This is required to increase the accuracy of the whole process. We also assume that there exists at least one appropriate hard ground shadow that is generated by a rigid body whose shape and position do not change during the time-lapse video.

As a future work, different smoothing and filtering techniques can be experimented to improve the results of the application of the proposed approach online. Soft ground shadows may be evaluated in order to track some other light sources that may have effect on the environment. Moreover, the shadows other than the ones on the ground may be used additionally to increase the accuracy. One other possible future extension is to evaluate the energy changes on the pixels that form the shadow tracked in the time-lapse video. This evaluation gives the opportunity to track the changes on the light intensity of the Sun as well as its position.

Bibliography

[1] J. Carmigniani, B. Furht, M. Anisetti, P. Ceravolo, E. Damiani, and M. Ivkovic, "Augmented reality technologies, systems and applications," *Multimedia Tools and Applications*, vol. 51, no. 1, pp. 341–377, 2011.

[2] A. Panagopoulos, C. Wang, D. Samaras, and N. Paragios, "Illumination estimation and cast shadow detection through a higher-order graphical model," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, pp. 673–680, June 2011.

[3] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan, "Estimating the natural illumination conditions from a single outdoor image," *International Journal of Computer Vision*, vol. 98, no. 2, pp. 123–145, 2012.

[4] Y. Liu, T. Gevers, and X. Li, "Estimation of sunlight direction using 3D object models," *IEEE Transactions on Image Processing*, vol. 24, no. 3, pp. 932–942, 2014.

[5] J.-F. Lalonde and I. Matthews, "Lighting estimation in outdoor image collections," in *Proceedings of the International Conference on 3D Vision*, vol. 1, (Tokyo, Japan), pp. 131–138, 2014.

[6] Y. Liu and X. Granier, "Online tracking of outdoor lighting variations for Augmented Reality with moving cameras," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 4, pp. 573–580, 2012.

[7] M. S. Andersen, T. Jensen, and C. B. Madsen, "Estimation of dynamic light changes in outdoor scenes without the use of calibration objects," *Proceedings*

of the *International Conference on Pattern Recognition*, vol. 4, pp. 91–94, 2006.

[8] G. Xing, Y. Liu, X. Qin, and Q. Peng, "On-line illumination estimation of outdoor scenes based on area selection for augmented reality," in *Proceedings of the 12th International Conference on Computer-Aided Design and Computer Graphics*, CADGRAPHICS '11, (Washington, DC, USA), pp. 439–442, IEEE Computer Society, 2011.

[9] T. Ikeda, Y. Oyamada, M. Sugimoto, and H. Saito, "Illumination estimation from shadow and incomplete object shape captured by an RGB-D camera," in *Proceedings of the 21st International Conference on Pattern Recognition*, ICPR '12, (Tsukuba, Japan), pp. 165–169, November 2012.

[10] L. Gruber, T. Richter-Trummer, and D. Schmalstieg, "Real-time photometric registration from arbitrary geometry," in *Proceedings of the 11th IEEE International Symposium on Mixed and Augmented Reality*, ISMAR '12, (Atlanta, GA, USA), pp. 119–128, November 2012.

[11] L. Gruber, T. Langlotz, P. Sen, T. Hoherer, and D. Schmalstieg, "Efficient and robust radiance transfer for probeless photorealistic augmented reality," in *Proceedings of the IEEE Virtual Reality*, VR '14, (Minneapolis, MN, USA), pp. 15–20, 2014.

[12] P. Lensing and W. Broll, "Instant indirect illumination for dynamic mixed reality scenes.," in *Proceedings of the 11th IEEE International Symposium on Mixed and Augmented Reality*, ISMAR '12, pp. 109–118, IEEE Computer Society, 2012.

[13] N. Neverova, D. Muselet, and A. Trémeau, "Lighting estimation in indoor environments from low-quality images," in *Proceedings of the 12th International Conference on Computer Vision*, ICCV '12, (Florence, Italy), October.

[14] J. Yoo and K. Lee, "Light source estimation for realistic shadow using segmented HDR images," in *Proceedings of the International Symposium on Ubiquitous VR* (D. Hong and S. Jeon, eds.), vol. 260 of *ISUVR '07*, CEUR-WS.org, 2007.

[15] J. Lopez-Moreno, S. Hadap, E. Reinhard, and D. Gutierrez, "Light Source Detection in Photographs," in *Congreso Espanol de Informatica Grafica* (C. Andujar and J. Lluch, eds.), CEIG '09, The Eurographics Association, 2009.

[16] J. Lopez-Moreno, E. Garces, S. Hadap, E. Reinhard, and D. Gutierrez, "Multiple light source estimation in a single image," *Computer Graphics Forum*, vol. 32, pp. 170–182, December 2013.

[17] K. Rohmer, W. Büschel, R. Dachselt, and T. Grosch, "Interactive near-field illumination for photorealistic augmented reality on mobile devices," in *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality*, ISMAR '14, pp. 29–38, September 2014.

[18] I. Arief, S. McCallum, and J. Y. Hardeberg, "Realtime estimation of illumination direction for augmented reality on mobile devices," in *Proceedings of the Color and Imaging Conference*, (Los Angeles, CA, USA), pp. 111–116, IS&T and SID, November 2012.

[19] P. Debevec, "Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography," in *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '98, (New York, NY, USA), pp. 189–198, ACM, 1998.

[20] M. Kanbara and N. Yokoya, "Real-time estimation of light source environment for photorealistic augmented reality," in *17th International Conference on Pattern Recognition*, ICPR '04, (Cambridge, UK), pp. 911–914, August 2004.

[21] K.-Y. K. Wong, D. Schnieders, and S. Li, "Recovering light directions and camera poses from a single sphere," in *Proceedings of the 10th European Conference on Computer Vision*, vol. 1 of *ECCV '08*, (Berlin, Heidelberg), pp. 631–642, Springer-Verlag, 2008.

[22] S. Lee and S. K. Jung, "Estimation of illuminants for plausible lighting in augmented reality," in *Proceedings of the International Symposium on*

Ubiquitous Virtual Reality, ISUVR'11, (Jeju-si, Republic of Korea), pp. 17–20, July 2011.

[23] K. Nishino and S. Nayar, "Eyes for relighting," *ACM Transactions on Graphics (Proceedings of SIGGRAPH '04)*, vol. 23, pp. 704–711, July 2004.

[24] I. Sato, Y. Sato, and K. Ikeuchi, "Acquiring a radiance distribution to superimpose virtual objects onto real scene," in *Modeling from Reality, The Springer International Series in Engineering and Computer Science*, vol. 640, pp. 137–160, 1998.

[25] J. D. Yoo and K. H. Lee, "Real time light source estimation using a fisheye lens with nd filters," in *Proceedings of the International Symposium on Ubiquitous Virtual Reality*, ISUVR '08, (Los Alamitos, CA, USA), pp. 41–42, IEEE Computer Society, 2008.

[26] S. R. Marschner and D. P. Greenberg, "Inverse lighting for photography," in *Proceedings of the IS&T/SID Fifth Color Imaging Conference*, pp. 262–265, 1997.

[27] I. Sato, Y. Sato, and K. Ikeuchi, "Illumination from shadows," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 3, pp. 290–300, 2003.

[28] S. B. Knorr and D. Kurz, "Real-time illumination estimation from faces for coherent rendering," in *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality*, ISMAR '14, pp. 349–350, 2014.

[29] K. Sunkavalli, W. Matusik, H. Pfister, and S. Rusinkiewicz, "Factored time-lapse video," *ACM Transactions on Graphics (Proceedings of SIGGRAPH '07)*, vol. 26, no. 3, Article No. 101, 10 pages, 2007.

[30] R. Zhang, F. Zhong, L. Lin, G. Xing, Q. Peng, and X. Qin, "Basis image decomposition of outdoor time-lapse videos.," *The Visual Computer*, vol. 29, no. 11, pp. 1197–1210, 2013.

[31] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan, "Webcam clip art: Appearance and illuminant transfer from time-lapse sequences," *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia '09)*, vol. 28, December 2009.

[32] X. Chen, K. Wang, and X. Jin, "Single image based illumination estimation for lighting virtual object in real scene," in *Proceedings of the 12th International Conference on Computer-Aided Design and Computer Graphics*, CADGRAPHICS '11, (Washington, DC, USA), pp. 450–455, IEEE Computer Society, 2011.

[33] A. Saxena, M. Sun, and A. Ng, "Make3D: Learning 3D scene structure from a single still image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 824–840, 2009.

[34] M. F. Tappen, E. H. Adelson, and W. T. Freeman, "Estimating intrinsic component images using non-linear regression," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2 of *CVPR '06*, (Washington, DC, USA), pp. 1992–1999, IEEE Computer Society, 2006.

[35] L. Shen, P. Tan, and S. Lin, "Intrinsic image decomposition with non-local texture cues," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR '08, (Anchorage, Alaska, USA), 2008.

[36] X. Jiang, A. J. Schofield, and J. L. Wyatt, "Correlation-based intrinsic image extraction from a single image," in *Proceedings of the 11th European Conference on Computer Vision*, vol. 4 of *ECCV '10*, (Berlin, Heidelberg), pp. 58–71, Springer-Verlag, 2010.

[37] E. Garces, A. Munoz, J. Lopez-Moreno, and D. Gutierrez, "Intrinsic images by clustering," *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)*, vol. 31, no. 4, pp. 1415–1424, 2012.

[38] A. Aman and U. Gudukbay, "Model based camera tracking for augmented reality," *submitted*.

[39] B. T. Phong, "Illumination for computer generated pictures," *Communications of the ACM*, vol. 18, no. 6, pp. 311–317, 1975.

[40] J. J. Moré, "The Levenberg-Marquardt algorithm: Implementation and theory," in *Numerical Analysis* (G. A. Watson, ed.), pp. 105–116, Berlin: Springer, 1977.

[41] J.-F. Lalonde, A. Efros, and S. Narasimhan, "Detecting ground shadows in outdoor consumer photographs," in *Proceedings of the European Conference on Computer Vision* (K. Daniilidis, P. Maragos, and N. Paragios, eds.), vol. 6312 of *ECCV '10*, pp. 322–335, Springer Berlin Heidelberg, 2010.

[42] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," in *Proceedings of ACM SIGGRAPH*, SIGGRAPH '07, (New York, NY, USA), ACM, 2007.

[43] Unity Technologies, "Unity - Game Engine." `http://unity3d.com/`. Accessed: 2015-07-15.

[44] MathWorks, "MATLAB - The Language of Technical Computing." `http://www.mathworks.com/products/matlab/`. Accessed: 2015-07-15.

[45] G. Bates, "Fast food shadows timelapse." `https://www.youtube.com/watch?v=mdhS6pds8VY`. Accessed: 2015-07-15.

[46] M. Davies, "Shadows timelapse." `https://www.youtube.com/watch?v=Lvhjbrr5GI8`. Accessed: 2015-07-15.

[47] T. Gazda, "Sun's shadow time lapse." `https://www.youtube.com/watch?v=3B7KLstUZbI`. Accessed: 2015-07-15.