Real-Time Illumination Estimation from Faces for Coherent Rendering

Sebastian B. Knorr[*] Daniel Kurz[†]

Metaio GmbH

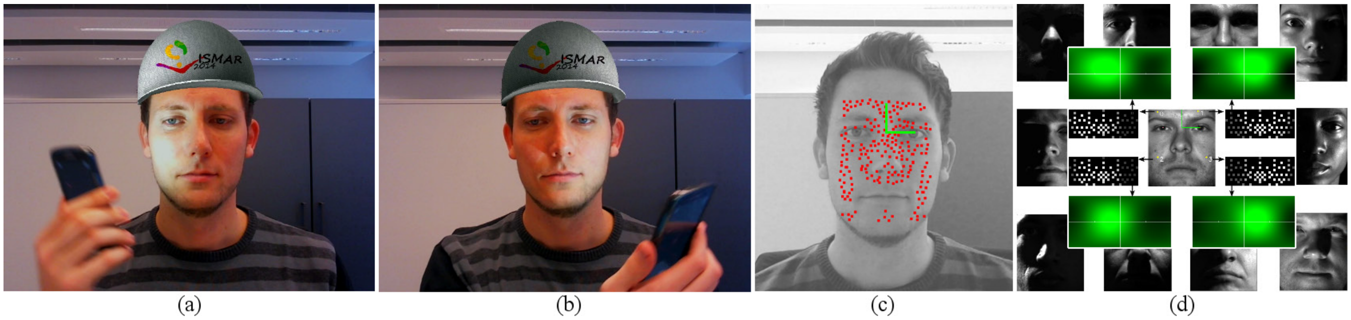

Figure 1: Our method enables coherent rendering of virtual augmentations (a,b) based on illumination estimation from the intensities of sample points in a face (c) and their radiance transfer functions learned from a dataset comprising of images of faces with known illuminations (d).

Abstract

We present a method for estimating the real-world lighting conditions within a scene in real-time. The estimation is based on the visual appearance of a human face in the real scene captured in a single image of a monocular camera. In hardware setups featuring a user-facing camera, an image of the user's face can be acquired at any time. The limited range in variations between different human faces makes it possible to analyze their appearance offline, and to apply the results to new faces. Our approach uses radiance transfer functions – learned offline from a dataset of images of faces under different known illuminations – for particular points on the human face. Based on these functions, we recover the most plausible real-world lighting conditions for measured reflections in a face, represented by a function depending on incident light angle using Spherical Harmonics.

The pose of the camera relative to the face is determined by means of optical tracking, and virtual 3D content is rendered and overlaid onto the real scene with a fixed spatial relationship to the face. By applying the estimated lighting conditions to the rendering of the virtual content, the augmented scene is shaded coherently with regard to the real and virtual parts of the scene. We show with different examples under a variety of lighting conditions, that our approach provides plausible results, which considerably enhance the visual realism in real-time Augmented Reality applications.

1 Introduction

The concept of Augmented Reality (AR) is based on a view of the real-world environment (commonly in form of a live video stream) which is combined with an overlay of virtual content in a spatial relationship to the real world. For many AR applications, the virtual content shall seamlessly integrate in the image of the real-world environment giving the user an immersive AR experience and the impression that the virtual content is actually placed within the real world. The ultimate goal is to combine the real and virtual scene parts in a *photo-realistic* way, where the user cannot distinguish between real and virtual parts. Lighting has a big impact on appearance and in order to make the augmented view more realistic, it is important to illuminate the virtual objects with the same lighting conditions visible in the real-world environment. Inconsistent illumination manifests for example in improper coloring as well as wrong positions of highlights and cast shadows and thereby disrupts the realistic impression.

Therefore the real-world lighting conditions must be acquired – preferably in real-time at the time of augmentation – to apply them to the virtual content. We present an approach to estimate lighting conditions from a single monocular image of the user's face in real-time. Based on the visual appearance of the face we estimate the incident light that led to the observed appearance. Our approach is particularly beneficial for use cases where virtual objects are augmented on the user's face or close to it. Common examples include virtual try-on of glasses, jewelry, or hats, as in the example shown in figure 1 (a,b). Different setups comprising a user-facing camera may take advantage of our method, including AR kiosks, web-based shopping applications, and handheld devices – such as smartphones and tablet PCs – for mobile AR experiences.

Our approach falls in the area of supervised machine learning and regression analysis, as we estimate the illumination based on known properties learned from training data. Based on a dataset of images of faces captured under known different illuminations, we learn in an offline process how different locations (i.e. sample positions) on the face reflect light towards the camera in relation to light incident on the face from the illumination in the environment. Based on this knowledge we can estimate the current lighting conditions in real-time from a single image of a face under arbitrary illumination. For any new previously unseen image of a face, we first perform a face detection to determine the coordinate system in which the sample positions on the face are defined, see figure 1 (c). We then query the intensities of the new image at those positions and use them to solve for the unknown illumination. The estimated lighting conditions are eventually used for coherent rendering of virtual objects superimposed on the image of the face. Thereby, we achieve visually plausible AR experiences that adapt to the real-world illuminations in real-time.

[*] e-mail:sebastian.knorr@metaio.com
[†] e-mail:daniel.kurz@metaio.com

IEEE International Symposium on Mixed and Augmented Reality 2014
Science and Technology Proceedings
10 - 12 September 2014, Munich, Germany
978-1-4799-6184-9/13/$31.00 ©2014 IEEE

2 Related Work

In one of the pioneer works about common illumination in the context of combining virtual renderings with images of the real world Nakamae *et al.* [25] demonstrate how the ratio between the intensity of sun and sky light can be determined from image intensities of directly illuminated points and points in the shadow. Common illumination between real and virtual objects is simulated by Fournier *et al.* [10] by calculating two global radiosity solutions based on a model of the real world – with and without added virtual content. The ratio between the two solutions is used to modify the intensity of the parts of the real world. Furthermore intensities of known light sources are recovered finding the linear combination of per light source radiosity solutions best fitting to an image. Debevec [7] uses the *difference* between two global illumination solutions (with and without virtual content) to calculate the interaction of light between the virtual and real objects. Within this *Differential Rendering* the scene is partitioned into three components: the (real) *distant scene*, the (real) *local scene*, and the *synthetic objects*. Similar to our approach, the distant scene only emits light, and is not influenced by the addition of synthetic objects. Local scene and synthetic objects in contrast influence each other in terms of light interactions and thus need to be modeled including geometry and material.

Different approaches exist for acquiring real-world illumination. One is to *directly measure* the incoming light for example by capturing images of a mirrored ball – a so called light probe – within the scene at the target position for the virtual content [7]. Sato *et al.* [32] instead use fisheye cameras to capture omni-directional images. Recently Meilland *et al.* [24] demonstrate dense visual SLAM based on a low dynamic range RGB-D camera for creating a 3D map of the scene including high dynamic range (HDR) recovery to illuminate virtual objects. Methods based on direct measurements of the environment deliver high quality results but always come with the burden of a separate acquisition process.

An alternative approach for acquiring the incident illumination, which we pursue in this paper, is to *estimate* lighting conditions from the appearance of illuminated parts of the scene within the view of the camera. This approach is also known as *Inverse Lighting* and was introduced by Marschner and Greenberg [23] to reconstruct lighting from a photograph and a 3D model of the pictured object to modify the image afterwards according to a new user-specified lighting configuration. This approach models incident light by uniformly distributed directional basis lights and finds the linear combination of the corresponding basis images (generated using the 3D model) that best matches the photograph. The re-lighting is demonstrated for a diffuse rigid object as well as for a human face. In contrast to this approach, we do not rely on a 3D geometry model of a face.

A theoretical framework for the general problem of inverse rendering, that is measuring rendering attributes like lighting and reflectance properties from images, is introduced by Ramamoorthi and Hanrahan [31]. They analyze the mathematical foundation of the reflected light field and show for a curved convex, homogeneous surface under distant illumination using Spherical Harmonics (SH) representations, that the reflected light field can be described as a convolution of lighting and surface material, so that inverse rendering can be seen as deconvolution. They also explain ill-conditioning in light estimation from a Lambertian surface compared to a mirror-like surface. Similar insights are also presented by Basri and Jacobs [4]. It is shown, that the set of all reflectance functions for diffuse objects lies close to a 9D subspace and images of a diffuse (convex) object under variable lighting can be represented using only 9 basis functions.

Various methods in the context of Augmented Reality (AR) use some kind of inverse lighting to recover information of the illumination. Some rely on *known objects with predefined geometry and reflectance properties* that have to be placed additionally within the scene, such as a ping pong ball or a planar marker rotated in front of the camera as proposed by Aittala [2]. Arief *et al.* [3] estimate the direction of one dominant light source based on the shadow contour cast by a cuboid shaped *3D AR marker*, which is simultaneously used for tracking. A conventional 2D square marker for tracking is combined with an attached small black mirror ball by Kanbara and Yokoya [17]. The reflections of the 8 brightest spots are used to estimate directions, colors and intensities of the light sources. The reflections on a planar specular surface have also been exploited to reconstruct the illumination by Jachnik *et al.* [15]. Gruber *et al.* [14] present an approach of inverse lighting for *arbitrary* scene geometry instead of relying on predefined known objects. They use an RGB-D camera for geometry reconstruction and simultaneous recovering of the incident directional light distribution. As in this paper, they represent lighting and radiance transfer functions using low order SHs. Their calculation of radiance transfer functions is however based on the depth input from the RGB-D camera combined with the assumption of fully diffuse objects and they do not recover light colors but assume white light.

An alternative to acquiring the original illumination is to acquire the resulting shading, e.g. parameterized by surface orientation and visibility of the hemisphere, and directly apply it during rendering. Calian *et al.* [6] employ 3D-printed shading probes, that consist of a white kernel showing the convolved incident light. The white kernel is partitioned by black walls into different spherical sections. One section shows the diffuse shading for light from a particular part of the hemisphere. All kernel parts of the shading probe must be captured, requiring the user to rotate the camera around the probe. Yao *et al.* [35] acquire diffuse shading depending on surface orientation represented by SHs focusing like us on a particular body part of the user, namely the hand, which they capture using a RGB-D camera.

Beyond the area of AR, active research has been done regarding illumination of the human face, particularly in the field of *relighting* i.e. rendering of faces under new illumination and/or poses. Debevec *et al.* [8] acquire the light reflected from a human face by capturing images of the same face under dense sampling of incident illumination directions using a so called *Light Stage* and construct a reflectance function in form of an image for each image pixel. From these functions they can directly create new images of the face in any form of illumination. The reflectance function corresponds to the radiance transfer function in here. Fuchs *et al.* [11] analyze spatially varying reflectance properties of a particular human face by taking photos in calibrated environments under different poses and up to seven point-light conditions. They estimate the geometry of the particular face using a 3D Morphable Model [5] and fit parameters of an analytic BRDF model for different regions in the face as well as a fine-grained locally varying diffuse term. This allows rendering under new poses and complex lighting conditions. Additionally based on facial features, they may map the acquired reflectance properties from one face onto another one. Nishino and Nayar [27] compute the environment map of the scene from the reflections of the surrounding world visible in the image of an eye and use the result for light estimation, face relighting as well as for reconstruction of facial geometry. Illumination of faces is also highly relevant in the area of *face recognition*, as lighting often has a big impact on the image of a human face beside characteristics of a particular face, interfering with the goal to determine the person's identity. For recognizing a face under variation in lighting Georghiades *et al.* [12] build the illumination cone (i.e. the set of images of an object in a fixed pose, but under all possible illumination conditions) for a particular face from seven images of the same face and pose under different lighting directions by reconstructing shape and albedo. From one image of an unknown face under unknown illumination Sim and Kanade [33] create new images under changed illumination for better face recognition using the standard Lambertian equation. Because this equation does not model shadows and specular reflections they extend the equation by an additive per pixel error term, capturing the error introduced by the simplification. They learn a statistical model for the normals as well as for the error term depending on the location on the face from a set of images of people under different known illumination directions. The incident light direction from the image is estimated based on the difference between the input image and each training image using a Gaussian weighted sum over the corresponding known light directions. For the images out of the training set itself, they demonstrate high accuracy on the recovered light direction. Based on a collection of 3D face scans Zhang and Samaras [36] create a statistical model for the illumination of the human face. They compute per pixel means and covariances of Gaussian distributions for the influence of the different SH basis functions and model illumination considering only the surface orientations of the 3D scans, thereby assuming a convex diffuse object. Afterwards based on images of faces under known lighting an additional error term for the statistical model is estimated, comprising deviations from the diffuse as well as the convex assumption. Based on this model, SH coefficients of the unknown illumination for a given face image are estimated using kernel regression. 3D models of faces are also used by Qing *et al.* [28] to create a multitude of images showing the influence of different SH basis functions on the faces, again considering only the surface orientations of the 3D models. Average images, obtained by PCA, from the set of the images for the influence of a particular SH basis function are used to estimate the unknown illumination for a given unknown face image. SH illumination of faces is also combined with a 3D Morphable Model of the face for face recognition by Yhang *et al.* [37]. Lee *et al.* [21] demonstrate that basis images for the image variation of human faces under variable lighting (in terms of a good representation for face recognition) can be directly generated using real images with a certain set of configurations of lighting directions. They compare their results to harmonic images of the face and to the illumination cone. As opposed to the above approaches, sparse sampling of intensities in the image of the face is sufficient for our approach, as we are only interested in the lighting conditions and not in re-lighting the image of the face itself. Instead we use the estimated illumination for rendering virtual objects coherently.

Having acquired the real-world illumination, photo-realistic images with combined virtual and real content need to be generated in real-time. Knecht *et al.* [20] for example present a method for approximating the global illumination combining differential rendering with instant radiosity by Keller [18]. Also the simulation of camera effects for the virtual content has an important impact on the coherent appearance of virtual and real parts, like shown by Klein and Murray [19], who model artifacts arising during the imaging process, for example distortions, chromatic aberrations, blur and noise. Sophisticated rendering methods such as these are out of the scope of this paper, which focuses on estimating the real-world lighting conditions.

3 Light Estimation from Faces

Throughout this paper, we explicitly focus on the appearance of the face of the user as illuminated part of the scene for estimating incident light. Various benefits of this specialization are discussed in section 3.1. We estimate light incident on the face based on light reflected from the face towards the camera. A separation of the reflected light and the incident light is provided in section 3.2. The correlation between incident light from a particular direction and the radiance reflected towards the camera can be described with a radiance transfer function (RTF), which is explained in section 3.3.

Our approach for estimating the distant illumination from an image of a face consists of an offline learning stage and a real-time light estimation. The offline stage (see section 3.4) uses a set of images of faces under different known directional illumination from *The Extended Yale Face Database B* [12, 21] to *learn the average RTFs* over different humans for a set of sample positions on the face. We model the RTFs as well as the incident light from the distant environment using real-valued Spherical Harmonics (SH) – orthonormal basis functions over the domain of directions. In the online stage (see section 3.5) we receive an image of the face of the user as input and *estimate the unknown illumination* based on the intensity values at the sample positions on the face and the corresponding RTFs identified in the offline learning stage.

3.1 Benefits of the User's Face for Light Estimation

For AR scenarios, relying on the appearance of the user's face to estimate illumination has a number of benefits compared to State-of-the-Art approaches, which are based on either generic environments or specific objects that explicitly need to be placed in the scene. Firstly the user is already part of the scene, so no extra geometry must be placed and the appearance of the scene is not influenced. Furthermore, our approach does not require capturing the environment beforehand but can immediately estimate the illumination once the face is visible in the camera. This is always the case when dealing with a user-facing camera (e.g. next to the display) so that the user does not need to pay attention to keep some special geometry for light estimation within the camera's field of view. In virtual fitting use cases based on kiosk, web or mobile applications, where the augmented objects are for example glasses, jewelry or hats, the face of the user is already within the image of the camera and it is located close to the augmentation, which is another benefit.

The geometry of a human face is also well suited for light estimation, as it contains roughly the whole range of surface orientations facing the camera. Most importantly, human faces have a limited range of variations between different individuals regarding for example geometry and reflectance properties. As a result, these properties can be modeled (manually or automatically from a multitude of different faces) in an offline pre-process, which enables using optimized algorithms based on valid assumptions and restrictions with regard to faces, that run more efficiently than generic approaches. Light estimation based on arbitrary scenes requires acquisition of their geometry, e.g. using RGB-D cameras [14], and most importantly, there is no warranty that the scene is suited for light estimation. A planar table, for example, would reveal only little information about the illumination.

Another problem with arbitrary and unknown geometry is the ambiguity between light and material. Acquiring material properties on the fly under unknown light conditions is quite difficult and typically hardly possible from a single image. Assumptions that would be invalid for arbitrary scene surfaces can be applied to faces, like a specific model for skin reflectance which constrains the physical problem of ambiguity between surface material and light intensity and color. Regions of the face particularly suited for estimating the illumination can be pre-learned or pre-defined and distinguished from other regions. Using faces also allows algorithms to fit a generic 3D face model [5] using a single image captured by a standard RGB camera. With the face of the user staying the same over time, this fitting only has to be done once, compared to arbitrary parts of the scene geometry which change while the user is moving through the scene.

3.2 Light Field Separation

The image we perceive from our environment is the light distribution as equilibrium solution arisen after multiple reflections of light. The human vision system does not observe the propagation of light but only the final result. The same is true for standard cameras.

For a particular wavelength at a particular point in time, the 5D plenoptic function $P(x, \vec{\omega})$ [1] returns radiance within a light field along a ray specified by its 3D location $x \in \mathbb{R}^3$ and 2D orientation defined by a unit vector $\vec{\omega} \in \mathbb{R}^3$.

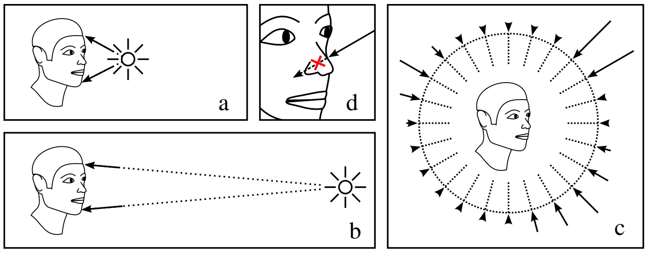

Figure 2: Light rays incident on an object have a stronger variation in direction for near light sources (a) than for distant ones (b). We model light incident from the distant scene as depending on direction only (c). Local surface points e.g. on the cheek may be occluded from incident light by other parts of the local scene e.g. the nose (d).

We assume that light is emitted only from an environment which is distant in comparison to the dimensions of the local object (e.g. the face) and we therefore consider two separate parts of the light field (similarly as in [7]), where the first part corresponds to the *distant scene* illuminating the *local scene*, and the second one corresponds to the light reflected from the *local scene*, which does not emit any light by itself.

Because the distant part is assumed to be located far from the local lit and displayed scene part, the parallax effect regarding incident light can be neglected for locations within this considered range of the local scene. The light incident from the distant scene thus depends on direction only and not on position. Figure 2 illustrates the concept behind. As long as the light source is close to the face (see figure 2 (a)), the direction of incident light from the light source varies quite strong between different positions on the face. With increasing distance (see figure 2 (b)) between the light and the lit object, this variation diminishes and incident light rays become more parallel. Light incident from a distant environment thus can be specified as a 2D function depending on incident direction only (see figure 2 (c)), which in the following is referred to as *directional distribution of incident light* $E(\vec{\omega}_i)$ (see figure 3). Note that $\vec{\omega}_i$ refers to the direction where light comes from and not where it is heading in this case.

The second considered part of the light field $R(x,\vec{\omega})$ represents light reflected at a surface point x of the local scene *into* the direction $\vec{\omega}$ and depends on E – the light incident from the distant scene – as well as on the material and geometry properties of the local scene. Light incident from the distant environment can be occluded for a particular point by another part of the local scene (see figure 2 (d)) resulting in cast shadow and light can be reflected from one local surface point onto another one. In the following we will have a closer look at the function modeling this correlation between the light field parts E and R, which we refer to as *Radiance Transfer Function (RTF)*.

3.3 Radiance Transfer Function

Mathematically the process of propagation of light can be formulated as an integral equation called *Rendering Equation* [16] :

$$L(x,\vec{\omega}) = L_e(x,\vec{\omega}) + L_r(x,\vec{\omega}) \tag{1}$$

$L(x,\vec{\omega})$ specifies the light – more precisely radiance – at a surface point $x \in \mathbb{R}^3$ *into* direction $\vec{\omega}$, which is composed of two parts: $L_e(x,\vec{\omega})$, radiance *emitted* at location x into direction $\vec{\omega}$ and $L_r(x,\vec{\omega})$, radiance *reflected* at location x into direction $\vec{\omega}$. L_e thereby directly corresponds to the light sources, while L_r can be further disassembled, referred to as *Reflection Equation*:

$$L_r(x,\vec{\omega}) = \int_{\Omega(x)} f_r(x,\vec{\omega}_i,\vec{\omega}) L_i(x,\vec{\omega}_i)(\vec{\omega}_i \cdot \vec{n}(x)) d\vec{\omega}_i \tag{2}$$

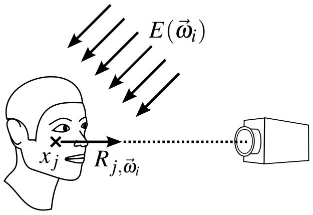

Figure 3: Light $E(\vec{\omega}_i)$ coming from the distant scene out of direction $\vec{\omega}_i$ incident on the face is transferred into radiance $R_{j,\vec{\omega}_i}$ leaving at point x_j towards the camera.

Radiance reflected at location x into direction $\vec{\omega}$ depends on incident radiance L_i at x, the surface orientation compared to the incident light and reflectance properties of the surface at x. Note that this formulation neglects subsurface scattering.

Let $\vec{n}(x) \in \mathbb{R}^3, \|\vec{n}\|_2 = 1$ be the outward-pointing normal vector specifying the surface orientation at x. The integral considers the incoming light at location x *from* all possible directions $\vec{\omega}_i \in \Omega(x)$. Thereby $\Omega(x)$ specifies the upper unit hemisphere with respect to the surface orientation $\vec{n}(x)$ at position x. The amount of incoming radiance $L_i(x,\vec{\omega}_i)$ from a particular direction $\vec{\omega}_i$ is scaled by the cosine of the angle between the direction and the surface orientation, thereby accounting only for the effective power incident on the unit area of the surface. The resulting irradiance is multiplied by the so called *Bidirectional Reflectance Distribution Function (BRDF)* [26] $f_r(x,\vec{\omega}_i,\vec{\omega})$, which specifies the ratio of *locally* reflected radiance into outgoing direction $\vec{\omega}$ to *locally* incident irradiance out of direction $\vec{\omega}_i$. This function depends on the material properties at the particular surface location x.

Because radiance along a ray does not change as long as light propagates through empty space, the radiance infalling at position x *from* direction $\vec{\omega}_i$ can be expressed as outgoing radiance *into* direction $(-\vec{\omega}_i)$ at the surface point visible from x in direction $\vec{\omega}_i$:

$$L_i(x,\vec{\omega}_i) = L(h(x,\vec{\omega}_i), -\vec{\omega}_i) \tag{3}$$

with $h(x,\vec{\omega}_i) \in \mathbb{R}^3$ returning the surface point visible from point x in direction $\vec{\omega}_i$. The rendering equation therefore can be written as:

$$L(x,\vec{\omega}) = L_e(x,\vec{\omega}) + \int_{\Omega(x)} f_r(x,\vec{\omega}_i,\vec{\omega}) L(h(x,\vec{\omega}_i),-\vec{\omega}_i)(\vec{\omega}_i \cdot \vec{n}(x)) d\vec{\omega}_i \tag{4}$$

Note that the unknown function L thereby occurs both inside and outside of the integral.

By applying the assumption that the light field can be separated into a distant part $E(\vec{\omega}_i)$ and a local part $R(x,\vec{\omega})$ we need to distinguish whether the local scene or the distant environment is visible from point x in direction $\vec{\omega}_i$. Figure 4 shows the division of the hemisphere $\Omega(x)$ into a set of directions where the distant environment is visible – marked as green – and the set where it is occluded by the local scene – marked as red.

For a point x on the surface of the local scene, the overall *reflected* light R into direction $\vec{\omega}$ then can be specified as:

$$\begin{aligned} R(x,\vec{\omega}) = & \int_{\Omega(x)} f_r(x,\vec{\omega}_i,\vec{\omega}) V(x,\vec{\omega}_i) \cdot E(\vec{\omega}_i) \cdot (\vec{\omega}_i \cdot \vec{n}(x)) d\vec{\omega}_i \\ & + \int_{\Omega(x)} f_r(x,\vec{\omega}_i,\vec{\omega})(1 - V(x,\vec{\omega}_i)) \cdot R(h(x,\vec{\omega}_i),-\vec{\omega}_i)(\vec{\omega}_i \cdot \vec{n}(x)) d\vec{\omega}_i \end{aligned} \tag{5}$$

with

$$V(x,\vec{\omega}) = \begin{cases} 1 & \text{if dist. env. visible at } x \text{ into direction } \vec{\omega} \\ 0 & \text{if dist. env. occluded at } x \text{ into direction } \vec{\omega} \end{cases} \tag{6}$$

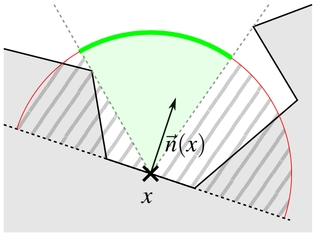

Figure 4: For parts of the directions of the hemisphere $\Omega(x)$ oriented along surface orientation $\vec{n}(x)$ at surface point x the distant environment is occluded (red), for other directions it is visible (green).

The recursion can be rewritten as an infinite series, a Neumann series with a linear operator $\mathbf{B}$ representing light transport by a single reflection step of light at a surface:

$$\begin{aligned} R &= \mathbf{B}(E+R) \\ &= \mathbf{B}(E+\mathbf{B}(E+R)) = \quad \cdots \quad = \sum_{i=1}^{\infty} \mathbf{B}^i(E) \\ &= \mathbf{T}(E) \end{aligned} \tag{7}$$

Operator $\mathbf{T}$ contains all the light transport – from direct illumination of the local scene ($\mathbf{B}(E)$), up to an infinite number of interreflections ($\mathbf{B}^{\infty}(E)$) within the local scene – and maps the distant part E of the light field to the reflected local part R. In function notation, let $T(x,\vec{\omega}_i,\vec{\omega})$ be the RTF corresponding to operator $\mathbf{T}$.

In general T is a real function over the domain of directions $\vec{\omega}_i$ from which light is incoming from the distant scene, positions x on the surface of the local scene and the directions $\vec{\omega}$ of the reflected light at x. For a surface point x of the local scene it specifies the ratio of outgoing radiance into direction $\vec{\omega}$ to radiance coming from the distant scene out of direction $\vec{\omega}_i$ incident on the *whole* local scene (see figure 3). Similar to the BRDF it also contains the material properties of the local scene, but it models the *global* light transport. It additionally includes the surface orientation as well as occlusions of parts of the distant environment at the surface location x by local geometry, and even the geometry and material properties of the whole local scene by interreflections of light within the local scene. According to [22], an RTF tabulates the linear response of a surface point in terms of *exit radiance* (R) to *source lighting* (E).

We are interested in the RTF at discrete points x_j on the surface of an object in the local scene, i.e. the face of the user. Considering a fixed pose (e.g. frontal head pose) in front of the camera, a particular surface point x_j also implicates a fixed direction $\vec{\omega}_j$ in relation to the coordinate system of the face from x_j towards the camera. For the image of the face, the brightness of the pixel corresponding to the projected x_j correlates to $R_j = R(x_j,\vec{\omega}_j)$, the reflected radiance at the surface point x_j into the direction of the camera $\vec{\omega}_j$. Let $T_j(\vec{\omega}_i) = T(x_j,\vec{\omega}_i,\vec{\omega}_j)$ be the RTF specifying the ratio of reflected light intensity at position x_j into direction $\vec{\omega}_j$ to light intensity coming from the environment from the particular direction $\vec{\omega}_i$.

Assuming that light is coming out of the distant scene from a single direction $\vec{\omega}_i$ only (see figure 3), we get the following equation for the reflected light at x_j into direction $\vec{\omega}_j$:

$$R_{j,\vec{\omega}_i} = T_j(\vec{\omega}_i)\cdot E(\vec{\omega}_i) \tag{8}$$

In real environments, light usually does not come from only a single direction, but from all directions with varying intensities. The environment then consists of a dense distribution of light intensities over the range of directions. The overall reflected light resulting from light incident from the distant scene from multiple directions is the sum over the reflected light intensities corresponding to each single incident light direction. For the continuous range of incident light directions the sum becomes an integral of the reflected light for incident light from the distant scene over all directions (specified as the unit sphere S^2). The integrand is the product of the RTF $T_j(\vec{\omega}_i)$ at the particular location x_j and the incoming light intensity $E(\vec{\omega}_i)$ from the distant scene both evaluated for the particular direction $\vec{\omega}_i$.

$$R_j = \int_{S^2} T_j(\vec{\omega}_i)\cdot E(\vec{\omega}_i)d\vec{\omega}_i \tag{9}$$

3.4 Learning the Impact of Illumination on the Appearance of Faces Offline

In the following we elaborate our training procedure to determine the RTF for particular points on a human face.

We restrict ourselves to a fixed pose, and without loss of generality we pick the frontal head pose. As shown in section 3.3 a fixed viewpoint induces a fixed reflection direction $\vec{\omega}_j$ towards the camera for a particular point x_j on the face. The RTF $T_j(\vec{\omega}_i)$ for point x_j only has the direction of incident light from the environment as independent variables. Given an RTF, the reflected light R_j depends on the specification of source lighting E only, see equation (9).

We want to support light estimation for arbitrary human faces, without a separate per person offline learning step. Therefore we want to determine for each sample position x_j the *average* RTF $T_j(\vec{\omega}_i)$ to approximate the different RTFs over different faces.

3.4.1 Input Training Data

We determine the RTF for a sample position x_j based on intensities of the corresponding pixel within images of different persons under different known directional illuminations. The images from the database used as input for the offline learning stage are a set of images of faces with *frontal head pose* from 38 human subjects under 64 different illumination conditions[12, 21].

Let F be the set of different faces with $f \in F$ specifying a particular face and K be the set of different directional illuminations with $k \in K$ specifying a particular distant illumination E_k containing only incident light from direction ω_k.

We define a coordinate system for an image of a face based on the positions of the eyes. Relative to this coordinate system, we (sparsely) select a set of sample positions x_j uniformly distributed within regions that are most likely skin regions for all different humans (like cheeks, forehead and nose). Let J be the set of selected samples with $j \in J$ specifying a particular sample. The same set J is used for all images. The positions of the eyes in the training images are labeled manually.

3.4.2 Per Person Albedo Factor

As a first assumption, we assume that for a particular location in the face – especially within the regions of the selected sample positions – the RTFs between different persons mainly vary by a uniform per person albedo term corresponding to the difference in the BRDF of the persons' skin. Therefore we first *normalize* the intensity of all training images of a person by dividing by the albedo of the respective person, which we determine by the median over the intensities of all sample points in the frontal lit image of the particular face. Simply scaling the RTFs however is a coarse not completely valid *approximation* for not fully convex and diffuse objects. Human skin exhibits a significant amount of glossy reflection and subsurface scattering. Also the geometry of a human face is not fully convex. We may want to improve this approximation in future work.

After compensating for the per person albedo factor, we assume that for a particular position in the face one RTF can be used to approximate the RTFs for all different persons.

Another similar approximation we make - with our database containing only grayscale images - is reusing the same RTF for different light frequencies just by scaling the RTF by an albedo factor specific to the frequency, i.e. color channel.

3.4.3 Spherical Harmonics Representations

We model all RTFs as well as incident light from the distant environment using real-valued Spherical Harmonics (*SH*) – orthonormal basis functions $Y_n(\vec{\omega})$ defined over the domain of directions. We use a linearized single index notation [34] with $n = \ell(\ell+1)+m$ where $\ell \in \{0,...,L\}$ specifies the degree or band of the SH basis function and $m \in \{-\ell,...,\ell\}$ the order within band ℓ. Please refer to [13] and [34] for a deeper insight. A real function $f(\vec{\omega})$ depending on direction – like an RTF or a distant illumination – can be approximated by a linear combination of SH basis functions. The linear combination is specified by the corresponding coefficients f_n for the SH basis functions $Y_n(\vec{\omega})$.

$$f(\vec{\omega}) = \sum_{n=0}^{\infty} f_n \cdot Y_n(\vec{\omega}) \approx \sum_{n=0}^{(L+1)^2-1} f_n \cdot Y_n(\vec{\omega}) \tag{10}$$

Our SH expansions will have maximum degree $L = 2$, which gives us 9 SH basis functions Y_n and corresponding coefficients f_n, that can be written as an SH coefficients vector $\hat{f} \in \mathbb{R}^9$ with $\hat{f} = (f_0, f_1, \cdots, f_8)$. The coefficients f_n can be determined by projecting the function $f(\vec{\omega})$ into the particular basis function $Y_n(\vec{\omega})$:

$$f_n = \int_{S^2} f(\vec{\omega}) \cdot Y_n(\vec{\omega}) d\vec{\omega} \tag{11}$$

Let $\hat{T}_j \in \mathbb{R}^9$ be the sought SH coefficients vector for the RTF $T_j(\vec{\omega}_i)$ at location x_j. T_j then is approximated by:

$$T_j(\vec{\omega}_i) \approx \sum_{n=0}^{8} \hat{T}_{j,n} Y_n(\vec{\omega}_i) \tag{12}$$

Let $\hat{E}_k \in \mathbb{R}^9$ be the SH coefficients vector for $E_k(\vec{\omega}_i)$ – the particular directional illumination k. E_k then is approximated by:

$$E_k(\vec{\omega}_i) \approx \sum_{n=0}^{8} \hat{E}_{k,n} Y_n(\vec{\omega}_i) \tag{13}$$

The images of the used database are each taken under light from one particular direction which is specified by azimuth and elevation angle. That means that the distant light field $E_k(\vec{\omega}_i)$ corresponding to an illumination k only contains light from this single direction $\vec{\omega}_k$. The integral in equation (11) thus becomes a direct evaluation of the basis function at this direction. We assume unit intensity.

$$\hat{E}_{k,n} = \int_{S^2} E_k(\vec{\omega}_i) \cdot Y_n(\vec{\omega}_i) d\vec{\omega}_i = \int_{S^2} \delta(\vec{\omega}_i - \vec{\omega}_k) \cdot Y_n(\vec{\omega}_i) d\vec{\omega}_i = Y_n(\vec{\omega}_k) \tag{14}$$

A directional light is locally defined in angular space, but contains all frequencies when defined in angular frequency space. An accurate representation by an SH expansion would need degree $L = \infty$. The limitation to $L = 2$ involves a coarse approximation.

The reflected light can be expressed as an integral of the product of RTF and particular distant illumination over all directions (eq. (9)). With both the RTF as well as the distant illumination expressed in SHs, we can exploit the orthonormal properties (see eq. (15)) of the SH basis functions:

$$\int_{S^2} Y_a(\vec{\omega}) \cdot Y_b(\vec{\omega}) d\vec{\omega} = \delta_{a,b} = \begin{cases} 1 & \text{if } a = b \\ 0 & \text{else} \end{cases} \tag{15}$$

When inserting the SH approximations from equations (12) and (13), the integral in equation (9) becomes a simple dot product of the SH coefficient vectors representing T_j and E_k.

$$R_{j,k} = \int_{S^2} T_j(\vec{\omega}_i) \cdot E_k(\vec{\omega}_i) d\vec{\omega}_i \tag{16}$$

$$\approx \hat{T}_j^\top \cdot \hat{E}_k \tag{17}$$

In the following we loosely write $=$ instead of $\approx$ also when referring to SH approximations.

3.4.4 System of Equations

Within the offline learning stage of the RTFs each sample position x_j is considered separately. Given a particular image i, with E_{k_i} being the distant illumination containing only incident light from direction ω_{k_i} and f_i being the particular face in image i. Let $I_{j,i}$ be the intensity of reflected light R_{j,k_i} (at surface point x_j into direction $\vec{\omega}_j$ by face f_i under illumination E_{k_i} measured by the intensity of the pixel corresponding to sample position x_j in image i) compensated by the albedo term of face f_i.

For each image we get an equation (17) between the *unknown* RTF T_j at position x_j in the face, the measured *known* reflected light intensity $R_{j,k}$ and the corresponding *known* illumination E_k.

From the set of images we can build a system of equations (18) for a particular surface point x_j and can calculate the least squares solution for the coefficients $\hat{T}_j$ of the RTF.

$$\begin{pmatrix} \hat{E}^\top_{k_{(i=1)}} \\ \hat{E}^\top_{k_{(i=2)}} \\ \vdots \\ \hat{E}^\top_{k_{(i=|K|\cdot|F|)}} \end{pmatrix} \cdot \hat{T}_j = \begin{pmatrix} I_{j,(i=1)} \\ I_{j,(i=2)} \\ \vdots \\ I_{j,(i=|K|\cdot|F|)} \end{pmatrix} \tag{18}$$

The offline stage results in recovered RTFs (each specified in SH coefficients $\hat{T}_j$) for the selected positions x_j in the face.

Figure 1 (d) illustrates how the RTF is evaluated for different sample positions. First the measured reflected intensities for one sample position are extracted from the multitude of images of the faces under different directional illumination. Then an RTF modeled by SH basis functions is fitted to the different measurements.

3.5 Online Illumination Estimation

In the online stage we receive an image of a (potentially unknown) face as input and thereof estimate the unknown directional distribution of incident light $E(\vec{\omega}_i)$.

Assuming that the sample positions within the image are already given, a system of equations similar to the one of equation (18) is built. In comparison to the offline learning process, where equations are collected *for one sample position x_j from a multitude of images*, this time equations for the directional distribution of incident light *within one image from the multitude of sample positions* are joined. The RTFs T_j for the different sample positions are known from the offline estimation step, but E is unknown. I_j is the intensity of reflected light R_j (at surface point x_j into direction $\vec{\omega}_j$ measured by the intensity of the pixel corresponding to sample position x_j) potentially compensated by an albedo term of the current face. This albedo term inhere leads to a scale of the estimated illumination. This is especially important when also estimating light color by making separate light estimations per color channel.

$$\begin{pmatrix} \hat{T}^\top_{j=1} \\ \hat{T}^\top_{j=2} \\ \vdots \\ \hat{T}^\top_{j=|J|} \end{pmatrix} \cdot \hat{E} = \begin{pmatrix} I_{j=1} \\ I_{j=2} \\ \vdots \\ I_{j=|J|} \end{pmatrix} \tag{19}$$

For this system of equations we calculate the least squares solution giving us the SH coefficients $\hat{E}$ of the directional distribution of incident light.

4 Face Tracking and Rendering of Virtual Objects

In order to build the system of equations (19) for an input image, the sample positions x_j defined on the face first must be projected onto pixel positions of the image. Therefore, and for positioning virtual content in a spatial relationship to the face, the face of the user must be tracked. Information on our face tracking and the projection of sample points x_j is given in section 4.1. With estimated illumination and determined pose we then render the augmented image. This rendering of the augmented scene (real plus virtual content) must run in real-time. We beforehand pre-compute the occlusion of the distant environment for the virtual content, as described in section 4.2. The real-time rendering uses the pre-computed data and combines it with the live estimated directional distribution of incident light in order to shade the virtual content coherently with the appearance of the real scene, see section 4.3.

4.1 Face Tracking

We use an image-based face tracking prototype as a black box. For an input image of a human face we obtain a 6DoF pose comprising 3D translation and 3D rotation. It is used for transforming the coordinate system for rendering virtual content as well as for projecting the sample positions x_j defined on the face onto pixel positions of the captured camera image. Note that in our prototype we beforehand projected the 2D sample positions onto a 3D face model for sake of simplicity. Figure 1 (c) shows the projected sample positions during live tracking. For now, our light estimation algorithm assumes close to frontal head poses in order to work properly.

4.2 Offline Pre-Computation for Rendering

For the shading of a virtual object, we currently only support direct lighting – no interreflections. The light coming from the distant environment is modeled using SHs up to 2nd degree. We pre-compute the influence of incident light from the distant environment on the virtual geometry as described in [34, 13]. For every vertex x of a virtual 3D model we calculate the influence c_n of each SH basis function Y_n modeling incident light on the intensity of the vertex:

$$c_n = \int_{\Omega(x)} V(x, \vec{\omega}_i) \cdot Y_n(\vec{\omega}_i)(\vec{\omega}_i \cdot \vec{n}(x)) d\vec{\omega}_i \qquad (20)$$

This influence c_n depends on the surface orientation at the vertex as well as on occlusions of the distant environment by the local scene itself (see figure 4). Besides the virtual object, the local scene hereby also contains a proxy geometry for the face which occludes parts of the environment, depicted in gray in figure 5. The calculation of the integral is done using Monte-Carlo integration by casting rays from the vertex position x randomly into all directions. The value of the SH basis function is evaluated for unoccluded directions, compensated by the cosine of angle between sample direction and surface orientation and summed up. For each vertex we thus obtain an SH coefficient vector $\hat{C} \in \mathbb{R}^9$, which will be supplied as per vertex attribute in the rendering stage. Figure 5 illustrates the coefficients of the SH basis functions over the surface of the model. Each image corresponds to the influence of one SH basis function.

For the proxy head model we additionally investigate pre-computing the differential change of the solutions with and without the virtual content, in order to simulate shadow cast from virtual content onto the real face. First results thereof can be found in figure S.4 in the *Supplemental Materials* as well as in the video.

4.3 Real-Time Rendering

Our implementation for the real-time rendering part is based on the Metaio SDK[1] using OpenGL and GLSL. Thanks to the image-based face tracking, virtual geometry is rendered in a fixed spatial relationship to the face.

The pre-computed SH coefficient vectors $\hat{C}$ from section 4.2 are supplied as per vertex attributes to the rendering stage. The estimated SH coefficients $\hat{E}$ of the directional distribution of incident light from section 3.5 are supplied in form of uniform arrays, with 9 coefficients each for red, green and blue light. The final irradiance for a vertex is determined by the dot products of $\hat{C}$ and $\hat{E}$. Note that SH coefficients pre-computed for the geometry and SH coefficients estimated for the lighting are already in the same coordinate system as long as the virtual geometry is fixed with regard to the face.

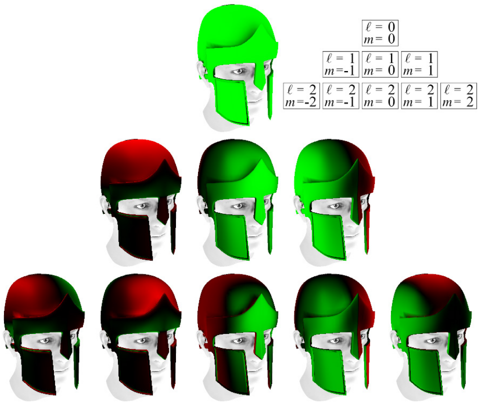

Figure 5: Pre-Computed Radiance Transfer (shadowed, no interreflections) for the rendering modeling the influence of each SH basis function on the particular point - green symbolizes positive, red negative influence, the brighter the greater the influence. The gray head is a proxy geometry used to incorporate occlusions by a real head.

5 Evaluations and Results

Below we evaluate our algorithm, compare estimated illuminations to ground truth and present visual results from live video sequences.

The per frame illumination estimation takes less than 1 ms for grayscale and less than 2 ms for RGB estimation on a Lenovo ThinkPad Helix i7-3667U (Windows 8.1 Pro) using a set of 294 sample positions.

5.1 Evaluation of Estimated Light Against Ground Truth

We first compare the primary light direction of our estimation with the ground truth light direction – both parametrized as SHs. Then we have a look at the visual qualitative impact of the differences between illumination estimation and ground truth.

5.1.1 Quantitative Results

In order to obtain a quantitative evaluation of the estimated illumination, we estimate illumination from images of faces under known directional illumination using the same database as for training. We divide the set of images from the used database beforehand into one part for training and a separate part for the evaluation.

For comparing the estimated illumination against ground truth, i.e. known azimuth and elevation of the directional light source, we extract the *optimal linear direction* [34] from the estimated illumination to approximate a directional light source. Note that this only utilizes the *linear* coefficients of the estimated lighting environment. The thus extracted values for azimuth and elevation from the estimated SH vector $\hat{E}$ representing the directional distribution of incident light are compared to the ground truth data for azimuth and elevation in figure 6.

Albeit there is some kind of imprecision, the estimations show a high degree of reliability. Note that the results contain all kind of images, including lighting under extreme angles. Also for lighting from above (elevation = 90°) there is a degree of freedom for the azimuth resulting in bad estimations at ground truth azimuth 0°.

Overall the estimation for the azimuth has a mean absolute error of 10.4° with a standard deviation of 20.6°. The estimation for the elevation has a mean absolute error of 8.2° with a standard deviation of 8.3°.

[1] http://www.metaio.com/sdk/

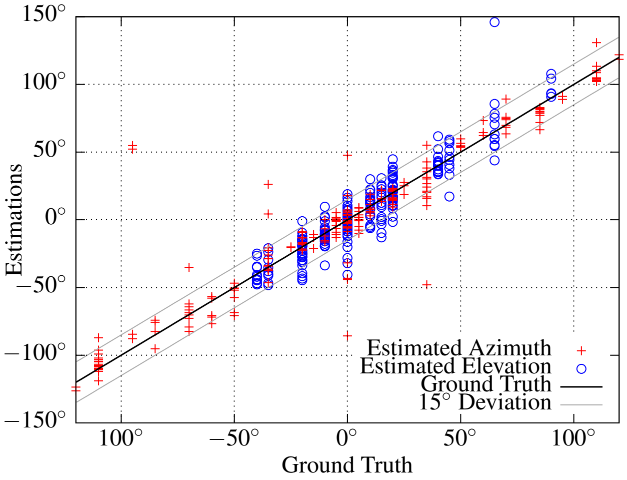

Figure 6: Comparison of estimations against ground truth.

We also examine the real angle between estimated and ground truth light direction by the dot product of the two direction vectors. This estimation has a mean absolute error of 12.3° with a standard deviation of 15.5°. Excluding the 2% of the estimations with an angular error above 75° the remaining estimations have a mean absolute error of 10.4° with a standard deviation of 7.1°.

5.1.2 Qualitative Results

Figure S.5 in the *Supplemental Materials* contains a grid of visualizations of light estimations for different faces and directional illuminations from the used database. Each estimation is illustrated by 6 parts. Part (a) displays the input image of a face used for estimating the incident illumination. Part (b) shows a latitude longitude image depicting the ground truth illumination resulting from projecting the known directional illumination into SHs. The brightness of a pixel in this image represents the light intensity out of the direction corresponding to the pixel. Green values symbolize positive values, while red values represent negative values. These physically non reasonable negative values for particular directions arise from the approximation of the directional light source by projecting it into the low dimensional space of SHs and cutting off higher frequencies of the SH expansion. Part (c) shows the same kind of latitude longitude image, however depicting the estimated illumination based on the image shown in part (a). Part (d) and (e) show renderings of a virtual face geometry using the illumination from part (b) and (c) respectively for better visual comparison. The renderings do not consider occlusions (accounting for the surface orientation only) and use full diffuse reflectance. Part (f) finally shows the difference image between the images from part (d) and (e).

The results demonstrate that the estimated illumination is in general comparable to the ground truth illumination also under harsh illumination from the side. The estimation however tends to overestimate intensities and compensates therefore using also higher negative intensities. We suspect, that using negative intensities allows the estimator to reproduce higher frequency effects visible in the input image like cast shadow and specularities, which could not be modeled by the low frequency RTFs. Also the estimated albedo for different faces seems to work reasonable well, visible in the similar scale of illumination of parts (d) and (e) in figure S.5.

Note that the evaluation on ground truth uses images which only contain one directional light source. For real-world applications, light is coming from all directions. A quantitative evaluation of our method on images with known environment light is part of our future work, e.g. by combining database images as in [36].

5.2 Qualitative Results on Webcam Sequences

Besides our results using the ground truth database, we ran our method on live video sequences captured with a webcam. In the following we provide qualitative results from sequences taken in multiple environments under varying illumination and with different users. As a result different faces, that are not part of the training dataset, act as a basis for illumination estimation in this case.

Figure 1 (a,b) shows two frames of a sequence where a user wears a virtual baseball cap on his head. To exaggerate changes in directional illumination, the person in this case uses the flashlight of a mobile phone to illuminate his own face. It is clearly visible, that the illumination used to render the virtual cap is consistent with the illumination apparent in the face, and therefore with the position of the flashlight. More examples for estimation of grayscale illumination are shown in figure 7 (a,b), where the light sources are lamps on a ceiling (a) and the sun in an outdoor scene (b).

Our approach is also capable of estimating the color of the incident light by estimating RGB (i.e. red, green, and blue) light individually. In figure 7 (c-e) we use a light source with controllable color to illuminate the face of a user. As can be seen particularly well in the insets showing the virtual white clown's nose (i.e. sphere) attached to the user's nose, the estimation of color succeeds and provides plausible illumination of the virtual contents.

Further examples and visual results can be found in figure S.2 and figure S.3 in the supplemental materials. We further show how our approach performs in real-time on image sequences in the supplemental video.

6 Conclusions and Future Work

In here we presented a method for estimating the illumination situation within a scene in real-time from the image of the user's face which allows coherent rendering of virtual and real parts in AR applications. The effectiveness of the method has been demonstrated in ground truth comparisons as well as under a variety of scenarios presented in image and video footage.

By discovering and exploiting the fact, that the face of the user is always within the scene and can be captured in many cases by a user-facing camera, we eliminated the use of a separate illumination estimation step which is needed in many State-of-the-Art approaches without us demanding any special hardware. The light estimation can be done inherently without the user even taking notice and runs on mobile devices in real-time.

Due to the limited range in variations between different human faces, we build a two-step algorithm extracting the expensive learning part into an offline process. RTFs for sample positions in the face representing the correlation between incident and reflected light are trained offline based on a plurality of images of faces under different known illumination. This knowledge is used for estimating the incident light from a single image of a face in real-time. The presented two-step algorithm could in future also be generalized for other objects than the face.

We intentionally designed our light estimation approach as simple as possible and we could show that it already provides pleasing results in a variety of cases. However, we believe that in future work some of the parts of our pipeline could be further improved.

Improving the Offline Training Stage For example the coordinate system for sampling intensities in the database images is defined only based on the eye positions and therefore does not model the differences in facial proportions between different humans. Using a more sophisticated model incorporating for example the position of the mouth or nose [33] or a full Morphable Model [11] could result in more accurate sample positions.

At the moment we are using the first 9 SH basis functions for the RTF and the estimated light. For diffuse lighting and convex diffuse geometries, when shading depends on surface orientation only, this

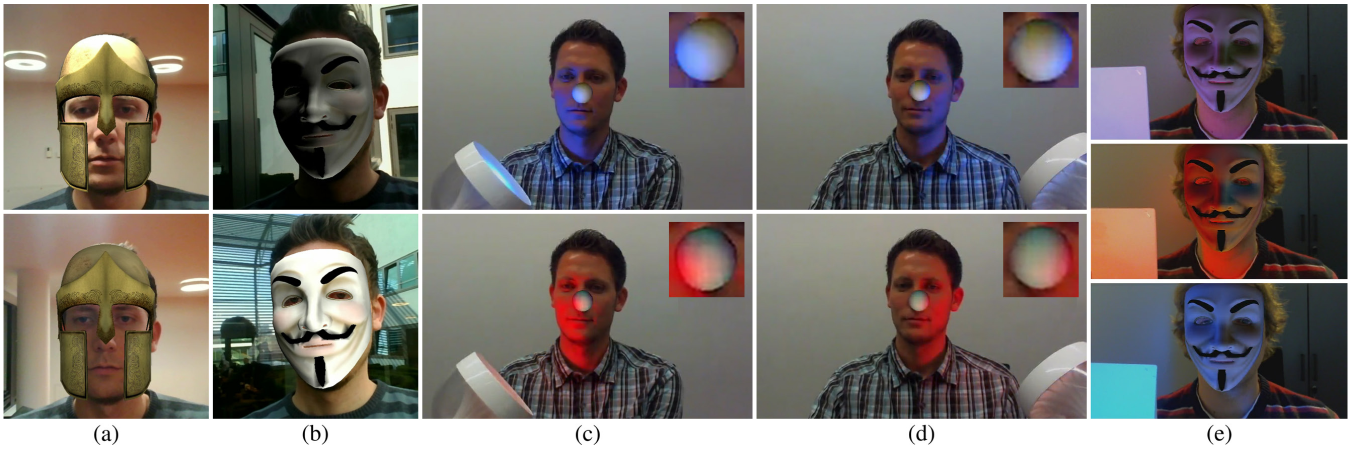

Figure 7: Results of grayscale (a,b) and RGB (c-e) illumination estimation and coherent rendering for different environments and illumination.

is sufficient as elaborated in [31, 30, 4]. Compared to other approaches, that often ignore cast shadows, we capture the occlusions within concave regions with our RTFs which thereby incorporate more information than only surface orientation. The SH approximation used for the RTFs however does not well model those high-frequent features. Resulting residuals when modeling variations by illumination using different numbers of eigenvectors have for example been evaluated by Epstein *et al.* [9]. Also the diffuse material assumption does not hold for faces. In order to capture and evaluate effects like cast shadows and glossy reflections, we plan to investigate the use of higher degrees of SHs or the use of some other appropriate function basis. On the other side Ramamoorthi [29] has analyzed the fact that a single image contains only roughly one half of all possible surface orientations – the front facing ones – and demonstrated that the variation within a single image of a convex diffuse object under arbitrary illumination can be even modeled by only 5 basis functions. He showed that orthogonality of the SH basis functions is no longer given for the restricted domain of visible surface orientations in one image. We want to investigate how far cast shadow and non-diffuse reflectance in faces as well as multiple images with different camera orientations reduce this phenomenon.

In the same context we plan to further evaluate the properties of the RTFs at different sample positions and measure which positions and distributions are well suited for light estimations. We used a first evaluation for reducing the number of sample positions. From an initial number of sample positions randomly positioned over the whole area of the face, we select the ones that have an absolute influence (coefficient) above a certain percentile for at least one SH basis function. Figure 8 shows the subsets of originally 512 uniformly distributed sample positions, that have an absolute influence (coefficient) above the 75-th percentile per SH basis function. We tested that approach reducing the number of sample positions from 512 to 294 (90-th percentile) without any significant decrease in accuracy. Another approach to reduce the number and evaluate the properties of sample positions (and thereby also determine the rank) would be a PCA over the SH vectors of the recovered RTFs. Groups of similar RTFs could be determined. This could also be used in a real-time per face optimization where we plan to select a subgroup of sample positions valid for a particular face. Sample positions exhibiting inconsistent intensities (e.g. macula, tattoo, beard, hair, geometric deviations) can be detected and excluded from the light estimation making the algorithm more stable for faces partially deviating from the learned model.

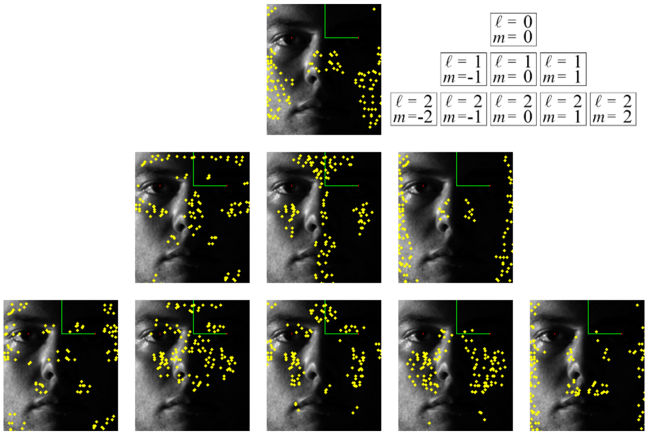

Figure 8: Sample positions with influence above the 75-th percentile; per SH basis function.

Our goal was to find *one* compact model for the RTFs that fits on different humans. We thus calculated the average RTFs over all different persons. We plan to investigate training separate RTF groups for different persons and potentially combine it with PCA. For online estimation the best fitting group could then be picked. We also presume that learned face properties well represent the tracked face. A user wearing a hat would for example violate this assumption. This condition could however also be learned from images.

The RTFs are estimated based on images with light coming (mainly) from in front of the user. Thereby the fact that light from behind the user has an influence close to zero for many sample positions, is not explicitly represented. Although the recovered RTFs (see figure 1 (d)) show plausible results, a more complete set of (also synthetic) training images with lighting from behind could make the estimation more reliable especially when a higher frequency approximation is used.

Improving the Online Estimation

The estimated real-world lighting conditions give plausible results when applied to virtual content for coherent rendering. However, our algorithm tends to also estimate negative light intensities from particular directions. As mentioned in section 5.1.2, negative intensities may arise from the low dimensional approximation of light sources by SHs. In here however negative lighting is used in combination with over estimated positive lighting to reproduce harsh variations in pixel intensities. This may lead to effects visible in figure 7 (c,d), where dominant red light on one side of the sphere leads to a lack in estimated red light components on the opposite side. The problem is enforced by the fact, that we only have observations of intensity for half of the possible surface orientations, leaving the optimization freedom in modeling back parts of the illumination. To resolve this problem, we plan to constrain the range of allowed solutions for the light estimation to prevalent positive intensities, for example using a convex optimization to enforce non-negative lighting as in [4].

For the images in the offline learning stage we assume some unit intensity of light. During online processing, we ignore the (nonlinear) camera response function and parameters such as exposure, contrast or color saturation settings. For physically meaningful estimations a radiometric calibration of the camera would be crucial. Our approach however mimics some camera effects by including them into the light estimation. An underexposed face for example leads to an estimation of low light intensity and to coherent *underexposure* of virtual content.

Another challenge, which is especially important for estimating colored (RGB) illumination, is an (online) albedo estimation for the user's face. Either active lighting using the camera flashlight or approaches based on cast shadows could be investigated.

Our current implementation estimates the incident light for each frame from a single image allowing the estimation to always be up-to-date even during rapid changes in illumination. Albeit for many cases this provides stable estimations we noticed in some scenarios high-frequent changes in the estimated illumination resulting in flickering augmentations. Temporal smoothing over multiple frames could eliminate this problem.

For the future we will tackle current limitations of the implementation as discussed above. Until now we focused on frontal facing faces only. We plan to extend our method to other poses. Depending on the head pose, different learned RTFs could be used or a model for the variation per viewing angle could be extracted.

Realistically showcasing products to the user will be a major requirement for successful AR kiosks and web- or app-based shopping applications. Estimating the present illumination is an important step for coherent rendering and is achieved by the method presented in here without posing any additional challenge to the user.

Acknowledgements

This work was supported in part by BMBF grant ARVIDA under reference number 01IM13001L.

References

[1] E. H. Adelson and J. R. Bergen. The Plenoptic Function and the Elements of Early Vision. *Computational Models of Visual Processing*, 1(2):3–20, 1991.

[2] M. Aittala. Inverse lighting and photorealistic rendering for augmented reality. *The Visual Computer*, 26(6-8):669–678, 2010.

[3] I. Arief, S. McCallum, and J. Y. Hardeberg. Realtime Estimation of Illumination Direction for Augmented Reality on Mobile Devices. In *Color and Imaging Conference*, 2012.

[4] R. Basri and D. W. Jacobs. Lambertian Reflectance and Linear Subspaces. *TPAMI*, 25(2):218–233, 2003.

[5] V. Blanz and T. Vetter. A Morphable Model For The Synthesis Of 3D Faces. In *Proc. SIGGRAPH*, 1999.

[6] D. A. Calian, K. Mitchell, D. Nowrouzezahrai, and J. Kautz. The Shading Probe: Fast Appearance Acquisition for Mobile AR. In *SIGGRAPH Asia 2013 Technical Briefs*, page 20, 2013.

[7] P. Debevec. Rendering Synthetic Objects into Real Scenes: Bridging Traditional and Image-based Graphics with Global Illumination and High Dynamic Range Photography. In *Proc. SIGGRAPH*, 1998.

[8] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar. Acquiring the Reflectance Field of a Human Face. In *Proc. SIGGRAPH*, 2000.

[9] R. Epstein, P. W. Hallinan, and A. L. Yuille. 5±2 Eigenimages Suffice: An Empirical Investigation of Low-Dimensional Lighting Models. In *Proc. Workshop on Physics-Based Modeling in Computer Vision*, 1995.

[10] A. Fournier, A. S. Gunawan, and C. Romanzin. Common Illumination between Real and Computer Generated Scenes. In *Graphics Interface*, 1993.

[11] M. Fuchs, V. Blanz, H. Lensch, and H.-P. Seidel. Reflectance from Images: A Model-Based Approach for Human Faces. *TVCG*, 11(3):296–305, 2005.

[12] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman. From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose. *TPAMI*, 23(6):643–660, 2001.

[13] R. Green. Spherical Harmonic Lighting: The Gritty Details. In *Archives of the Game Developers Conference*, 2003.

[14] L. Gruber, T. Richter-Trummer, and D. Schmalstieg. Real-Time Photometric Registration from Arbitrary Geometry. In *Proc. ISMAR*, 2012.

[15] J. Jachnik, R. A. Newcombe, and A. J. Davison. Real-Time Surface Light-field Capture for Augmentation of Planar Specular Surfaces. In *Proc. ISMAR*, 2012.

[16] J. T. Kajiya. THE RENDERING EQUATION. In *ACM Siggraph Computer Graphics*, volume 20, 1986.

[17] M. Kanbara and N. Yokoya. Real-time Estimation of Light Source Environment for Photorealistic Augmented Reality. In *Proc. ICPR*, 2004.

[18] A. Keller. Instant Radiosity. In *Proc. SIGGRAPH*, pages 49–56, 1997.

[19] G. Klein and D. W. Murray. Simulating Low-Cost Cameras for Augmented Reality Compositing. *TVCG*, 16(3):369–380, 2010.

[20] M. Knecht, C. Traxler, O. Mattausch, and M. Wimmer. Reciprocal Shading for Mixed Reality. *C&G*, 36:846–856, 2012.

[21] K.-C. Lee, J. Ho, and D. J. Kriegman. Acquiring Linear Subspaces for Face Recognition under Variable Lighting. *TPAMI*, 27(5):684–698, 2005.

[22] X. Liu, P.-P. Sloan, H.-Y. Shum, and J. Snyder. All-Frequency Precomputed Radiance Transfer for Glossy Objects. In *Proc. EG*, 2004.

[23] S. R. Marschner and D. P. Greenberg. Inverse Lighting for Photography. In *Color and Imaging Conference*, 1997.

[24] M. Meilland, C. Barat, and A. Comport. 3D High Dynamic Range Dense Visual SLAM and Its Application to Real-time Object Relighting. In *Proc. ISMAR*, 2013.

[25] E. Nakamae, K. Harada, T. Ishizaki, and T. Nishita. A Montage Method: The Overlaying of The Computer Generated Images onto a Background Photograph. In *ACM SIGGRAPH*, volume 20, 1986.

[26] F. Nicodemus, J. Richmond, J. Hsia, I. Ginsberg, and T. Limperis. Geometrical Considerations and Nomenclature for Reflectance. *Final Report National Bureau of Standards*, 1, 1977.

[27] K. Nishino and S. K. Nayar. Eyes for Relighting. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 704–711, 2004.

[28] L. Qing, S. Shan, and W. Gao. Eigen-Harmonics Faces: Face Recognition under Generic Lighting. In *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, 2004.

[29] R. Ramamoorthi. Analytic PCA Construction for Theoretical Analysis of Lighting Variability in Images of a Lambertian Object. *TPAMI*, 24(10):1322–1333, 2002.

[30] R. Ramamoorthi. Modeling Illumination Variation with Spherical Harmonics. *Face Processing: Advanced Modeling Methods*, pages 385–424, 2006.

[31] R. Ramamoorthi and P. Hanrahan. A signal-processing framework for inverse rendering. In *Proc. SIGGRAPH*, 2001.

[32] I. Sato, Y. Sato, and K. Ikeuchi. Acquiring a Radiance Distribution to Superimpose Virtual Objects onto a Real Scene. *TVCG*, 5(1):1–12, 1999.

[33] T. Sim and T. Kanade. Illuminating the Face. Technical Report CMU-RI-TR-01-31, Robotics Institute, 2001.

[34] P.-P. Sloan. Stupid Spherical Harmonics (SH) Tricks. In *Game Developers Conference*, volume 9, 2008.

[35] Y. Yao, H. Kawamura, and A. Kojima. The Hand as a Shading Probe. In *ACM SIGGRAPH 2013 Posters*, page 108, 2013.

[36] L. Zhang and D. Samaras. Face Recognition Under Variable Lighting using Harmonic Image Exemplars. In *Proc. CVPR*, 2003.

[37] L. Zhang, S. Wang, and D. Samaras. Face Synthesis and Recognition from a Single Image under Arbitrary Unknown Lighting using a Spherical Harmonic Basis Morphable Model. In *Proc. CVPR*, 2005.