Augmenting Physical Avatars Using Projector-Based Illumination

Amit Bermano[1,2], Philipp Brüschweiler[1,2], Anselm Grundhöfer[1], Daisuke Iwai[3], Bernd Bickel[1], Markus Gross[1,2]

[1]Disney Research Zurich [2]ETH Zurich [3]Osaka University

Figure 1: *Our system allows augmentation of a physical avatar (left) with projector-based illumnation, significantly increasing its expressiveness.*

Abstract

Animated animatronic figures are a unique way to give physical presence to a character. However, their movement and expressions are often limited due to mechanical constraints. In this paper, we propose a complete process for augmenting physical avatars using projector-based illumination, significantly increasing their expressiveness. Given an input animation, the system decomposes the motion into low-frequency motion that can be physically reproduced by the animatronic head and high-frequency details that are added using projected shading. At the core is a spatio-temporal optimization process that compresses the motion in gradient space, ensuring faithful motion replay while respecting the physical limitations of the system. We also propose a complete multi-camera and projection system, including a novel defocused projection and subsurface scattering compensation scheme. The result of our system is a highly expressive physical avatar that features facial details and motion otherwise unattainable due to physical constraints.

Keywords: projector-camera systems, animatronics

1 Introduction

Bringing virtual characters to life is one of the great challenges in computer graphics. While there were tremendous advancements in capturing, animating, and rendering realistic human faces in the past decade, displaying them on traditional screens conveys only a limited sense of physical presence. Animatronic figures or robotic avatars can bridge this gap. However, in contrast to virtual face models, reproducing detailed facial motions on an animatronic head is highly challenging due to physical constraints. Although steady progress in creating highly sophisticated robotic heads that strive to recreate convincing facial motions can be observed, for example those in Disney World's Hall of Presidents or "Geminoids" [Nishio et al. 2007], these achieve only limited expressiveness when compared to a real human being.

Our goal is to significantly increase the expressiveness of such figures, and to allow to animate them and controlling their motion and appearance easily, by adding additional degrees of freedom with projected shading. An animatronic head consists of a deformable skin attached to an underlying rigid articulated structure. The appearance is determined by the material of the skin and its static texture. The articulated structure is driven by a set of motors, and their motion range determines the expressiveness of the figure. While adding additional mechanical components to extend the degrees of freedom would be an obvious choice, in practice this is often prohibitive due to the lack of space inside the head and the extensive cost. Instead, we suggest projected shading to obtain dynamic control of the appearance, and emulate expressive motion and appearance using a combination of low-frequency motion of the animatronic head and high-frequency shading.

In this paper, we present a two-scale model for representing facial motion tailored to animatronic heads, embedded in a multi-projection system. Low-frequency motions that can be reproduced by the physical head are represented as control parameters of actuators. High-frequency details and subtle motions that cannot be reproduced are emulated in texture space. In practice, we face the challenge that the mechanical motion range of the robotic head is significantly smaller than that of a human. However, the formation of facial details is strongly correlated to the underlying low-frequency motion. Given an arbitrary performance capture sequence, a naive baking of dynamic facial details into texture space would violate this correlation, due to the limited mechanial motion range. The robotic head would stop moving when reaching its limit, while the original input data would still contain motion and induce formation of facial details. We observed that this leads to visual artifacts. We therefore propose an efficient spatio-temporal method for decomposing the motion in gradient space, ensuring that

we can reproduce the visual appearance of the sequence as close as possible while maintaining the correlation of low-frequency physical motion and formation of facial details. Using a multi-projector system, we then are able to convincingly and accurately replay the input animation.

More specifically, we start by acquiring a dense performance capture sequence of a person. First, we determine initial control parameters of the animatronic head that most closely resembles the target expression and acquire its detailed geometry for each frame. We then establish dense correspondence between the target performance and the performance of the animatronic head. Subsequently, we decompose the input performance into low-frequency animatronic head motion and dynamic high-frequency shading details. Given the dense correspondence, we perform a space-time optimization that maps the input performance to the constraint motion gamut of the robotic head. Subsequently, we embed the high-frequency shading information on the robotic head geometry such that the low-frequency details conform in both performances.

Furthermore, we present a complete multi-camera and -projector system, allowing efficient optimization of the projection quality in terms of focus and contrast. Defocused projections and subsurface scattering lower the possibilities to reproduce high-frequency shading on the animatronic head. To maximize the overall contrast and focus, we present a model-based multi-projector optimization step to improve the final image quality considering physical light drop-off, smooth blending in overlapping regions, projection defocus, and subsurface-scattering. The optimization is carried out by carefully analyzing and modeling the required defocus and subsurface scattering properties independently of the actual pose of the animatronic. This has the advantage that it, in contrast to camera-based approaches, is independent of a particular viewing position and can be easily adapted to arbitrary animatronic poses without exhaustive per-frame data acquisition.

We implemented a prototype and demonstrate several results with our system. In all our results one can observe that our approach significantly increases the expressiveness of the animatronic head. We also show how our system can be used for artistic effects such as aging of faces, an application that would not be possible without projector-based shading.

2 Related Work

Projected Avatars. Animated humanoid robots, called animatronics, are an old field of research. While currently high-quality animatronics exist that have a quite natural appearance [Ishiguro 2006; Nishio et al. 2007; Bickel et al. 2012], their movements and expressiveness are still limited and thus, while they appear almost like real humans, they are positioned in the deep dip of the uncanny valley. To overcome this problem, several research groups tried to use spatially varying illumination provided by projectors to superimpose the humanoid's face with dynamic textures to give it a more natural and dynamic appearance [Lincoln et al. 2009; Moubayed et al. 2012; Misawa et al. 2012; Kuratate et al. 2011]. First described as a generic principle in [Raskar et al. 2001], these approaches use a uniform white generic face geometry and apply projective texture mapping to superimpose colors and texture. While some of the heads are able to move rigidly, they are still significantly limited in their physical motion range. In contrast to those, we project onto a dynamic animatronic head having flexible, pigmented silicone skin. Thus, the physical head alone already enables limited non-rigid movement and the projection is employed for adding detailed shading on top.

Performance Transfer. Acquiring the expression of real faces and applying them to computer-generated models is a central component for creating lifelike performances [Havaldar et al. 2006]. A common method of performance transfer is encoding facial motion as a linear combination of target shapes and transferring the weights. The basis shapes can represent facial action units based on the facial action coding system [Ekman and Friesen 1977] or learned from data [Chuang and Bregler 2002; Blanz and Vetter 1999; Li et al. 2010]. For non-rigid mapping of the source performance to the target model, alternatively the deformation field can be directly transferred by establishing dense correspondence [Noh and Neumann 2001; Li et al. 2008]. Common to these remapping techniques is that there is a static mapping between source and target expression. However, as the motion gamut of the animatronic head is very limited, we desire a dynamic, temporally local compression. Inspired by the observation that the source and target movements should be similar, Seol et al. [2012] present a space-time facial animation re-targeting approach, interpreting movement as derivative in time and formulating the re-targeting problem as a Poisson equation. In our setup, we represent the motion of the robotic head in the constraint space of control parameters of the head [Bickel et al. 2012] and compute dense correspondences for transferring facial details and spatio-temporal optimization of the animatronics' head motion. In contrast to Seol et al. [2012], we do not optimize for global blend shape weights, but instead re-time the constrained coarse motion of the animatronic head to match the input motion as well as the correlation between coarse motion and the formation of facial details. In our approach, we make the important conceptual contribution that facial pose and facial details are performance-dependent and not statically coupled. This allows us to significantly increase the expressiveness of the projection-enhanced avatar.

Light-Transport-Based Projection Image Compensation Algorithms. The usage of projectors to change or enhance surface appearance has been an active research area for more than one decade. In [Wetzstein and Bimber 2007], a light-transport-based radiometric compensation method is described that extends local methods (cf. [Bimber et al. 2007] for an overview) to compensate for global illumination effects, such as defocus, refractions, diffuse, and subsurface-scattering for a particular camera view within the bounds of the capabilities of the used projector and camera. A quad-tree-based light transport measurement [Sen et al. 2005] is used, which takes up to several hours and thus is not practical for dynamic surfaces. This idea was recently extended in [Law et al. 2011; Aliaga et al. 2012] to generate a high-quality multi-projector compensation. Projector-camera systems can also be used to directly estimate the defocus of the projected pixels. If this defocus is measured and modeled as a point spread function (PSF) between the projector and the camera, an adapted image can be calculated which, up to a certain extent, compensates the projection defocus to make the image appear less blurred. Several approaches use a camera to evaluate the defocus on the surface and apply image filtering to generate a compensation image [Zhang and Nayar 2006; Oyamada and Saito 2008]. Multiple projectors were used in [Bimber and Emmerling 2006] to generate a blended projection with pixel contributions from several projectors to minimize the defocus. As this approach requires overlapping projections, the overall contrast of the system is reduced because of summed black intensity contributions. In [Grosse et al. 2010], a programmable aperture was integrated into the projector to generate a content-optimized image deconvolution. All compensation approaches so far, however, correct the defocus for a specific camera view and don't consider the oblique blur that is generated by a surface point that is not parallel to the camera's image plane. This effect was considered in [Nagase et al. 2011], which uses multiple projectors together with known geometry to calculate the camera-independent pixel contri-

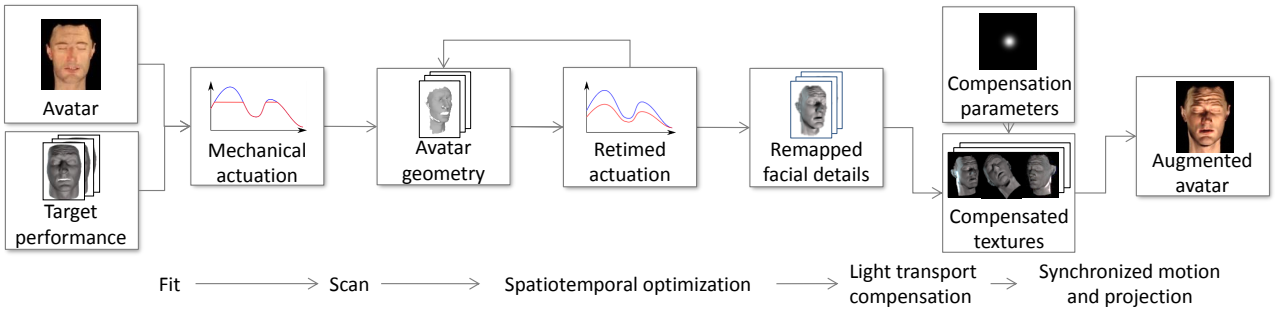


Figure 2: *Overview of the processing pipeline. A target performance drives the animatronics actuation, which is scanned by the system. Based on this data, the actuation parameters of the head are remapped to match the dynamics of the target performance. Next, the target performance is remapped onto the re-timed performance and its high-frequency details are embedded as colors. The sequence is then rendered from the calibrated projectors' point of view and gobally optimized to compensate for light drop-off, defocus, and subsurface scattering. Finally the resulting images are projected onto the animated animatronic head.*

bution. Our method applies a camera-independent compensation differently: by analyzing and modeling the system's defocus properties for a specific projection volume independent of the dynamic projection surface. These precomputed parameters are looked up for the given surface geometry and, in combination with a description of its subsurface scattering, are used to globally optimize the projection images for all used projectors.

3 Overview

Our approach on augmenting physical avatars using projector-based illumination starts by acquiring a source performance. For each input frame independently, we optimize for the animatronic head's actuation parameters that best resemble the input motion in simulation. Our goal is then to register the animatronic head to our projection-camera system, acquire information about its deformation behavior and subsurface-scattering as well as projector defocus to model the multi-projector light transport, perform a spatiotemporal decomposition and optimization of the head's motion and its texture to reproduce the desired facial performance, and finally, to reproduce the performance based on synchronized motion of the physical head and projection. An overview of the processing pipeline is given in Figure 2.

4 Performance Remapping

As input to our system we use a facial animation sequence that was captured using the system of Beeler et al. [2011]. It provides a detailed mesh sequence with explicit temporal correspondence. Our avatar is a proprietary animatronic head developed by Walt Disney Imagineering. It is driven by electric motors and features 13 parameters to control the actuation of the skin. We treat the underlying mechanical structure as a black box and use a finite-element-based optimization approach to determine the parameters for matching the deformation of the skin to each frame of the input sequence in a least-squares sense as described in Bickel et al. [2012]. We then place the animatronic head in our projector-camera system and acquire and register its dense performance as described in the following subsection. Furthermore, we then re-time its performance, as described in Section 4.3.

4.1 Geometry Acquisition

Accurate projection and remapping requires an accurate 3D representation of the physical avatar.

Acquisition setup. In order to enable the system to be self-contained once it is deployed, we used five calibrated cameras to capture structured light patterns for projector calibration, 3D reconstruction, and defocus estimation. The complete setup is depicted in Figure 3. The cameras are geometrically calibrated using a standard checkerboard-based calibration technique [Zhang 2000]. A series of structured light patterns, consisting of gray codes and binary blobs, is used to get a sub-pixel accurate mapping from camera to projector pixels. We then generate a medium-resolution 3D point cloud $\mathcal{P}_n$ for each frame $n = 1...N$ of the animatronic head's performance as described in [Hartley and Zisserman 2004]. Using direct linear transformation with non-linear optimization and distortion estimation enables an accurate calibration of the projectors. While the data provided by the scans is relatively accurate and represents the motion of the animatronic head well, it is incomplete in terms of both density and coverage: Regions that are not visible to more than one camera (due to occlusion or field of view) are not acquired at all, or yield a sparse and less accurate distribution of samples. Instead of adding more cameras to the system, we opted to scan the neutral pose once before deployment with a high-quality scanner [Beeler et al. 2011], and then to complete the missing data using non-rigid registration.

Non-rigid registration. Given the acquired point-clouds $\mathcal{P}_n$, we generate a complete detailed mesh sequence $\mathcal{M}_n$, using the high-quality scan of the neutral pose (denoted by $\mathcal{N}$). We achieve this by deforming $\mathcal{N}$ to match the point-cloud $\mathcal{P}_n$ in all high-confidence regions. For this, we first convert the point-cloud $\mathcal{P}_n$ to a manifold mesh $\hat{\mathcal{P}}_n$, by employing Poisson reconstruction [Kazhdan et al. 2006]. Using a similarity matching criterion combining distance, curvature, and surface normal as recommended in Tena et al. [2006], we then automatically find correspondences between $\hat{\mathcal{P}}_n$ and $\mathcal{N}$. The aforementioned process yields semantically plausible correspondences only for relatively small variations between meshes. Therefore, we use an incremental tracking process. For each frame n with corresponding acquired point-cloud $\mathcal{P}_n$, we use $\mathcal{M}_{n-1}$ as the high-quality mesh for the non-rigid registration step, assuming that the motion performed between two consecutive frames is sufficiently small. Using these correspondences, we then deform $\mathcal{N}$ to obtain a deformed mesh $\mathcal{M}_n$ that matches $\mathcal{P}_n$ using linear rotation-invariant coordinates [Lipman et al. 2005].

4.2 Actuator Control and Re-timing

We employ the physically based optimization method proposed by Bickel et al. [2012] to initially compute the animatronic actua-

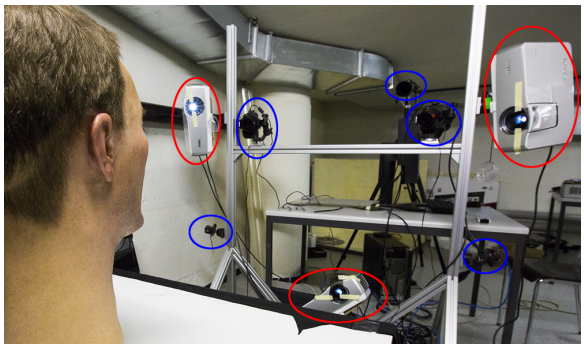


Figure 3: *Hardware setup: 5 cameras (blue) and 3 projectors (red) were used to reconstruct and illuminate the animatronic's face.*

tion control. This method matches the deformation of the skin to each frame of the target sequence individually. As the animatronic head's range of motion is much more limited than the target performance, the resulting motion follows the target one as long as it can, and remains stationary once the target motion is out of range. Projecting the target sequence in such a case results in textures that continuously present motion while the animatronic avatar does not. In practice, this results in significant visual artifacts. We therefore suggest augmenting the actuation by taking dynamics into consideration, and not only the poses of the performance. Figure 4 exhibits a result of the process applied on the eyebrows-raising sequence. The graph shows how the resulting motion resembles the target one in terms of dynamics more than actual deformation, while the images illustrate the effects the process has on the performance itself.

Temporal optimization. As our actuated performance was created using physically based simulation and the mapping between actuation parameters and resulting skin deformation is non-linear, we chose to adapt the timing of the existing performance instead of creating a new one, assuming linear behavior only between adjacent frames. In other words, given a sequence consisting of N frames, we wish to create a new sequence of the same length, with each frame being a linear blend of two adjacent frames of the original motion. We start by analyzing the temporally coherent mesh sequence for the actuated performance, as described in Section 4.1, $\mathcal{M}_n, n = 1..N$, along with its correspondence to the target performance $\mathcal{T}_n, n = 1..N$. Denoting the re-timed mesh sequence as $\hat{\mathcal{M}}_n, n = 1..N$, we represent it by a vector $\tau \in [1..N]^N$ such that $\tau_n \in \tau$ defines $\hat{\mathcal{M}}_n = \mathcal{M}_{\lfloor \tau_n \rfloor} \cdot \alpha + \mathcal{M}_{\lceil \tau_n \rceil} \cdot (1-\alpha), \alpha = (\tau_n - \lfloor \tau_n \rfloor)$. Using the error term discussed next, we wish to find a vector τ that minimizes the error between the target performance $\mathcal{T}_n$ and the augmented actuation frames $\hat{\mathcal{M}}_n$ induced by τ. In addition, we constrain τ to be temporally consistent such that each element $\tau_n \in \tau$ respects $\tau_n < \tau_{n+1}$. We employ constrained non-linear interior-point optimization to find the desired performance.

Error term. It has been shown that matching motion in the gradient space implies matching its dynamics instead of its pose and enhances realism [Seol et al. 2012]. However, as we do not have a linear face space, applying this principle directly to our case yielded undesirable results due to local minima. In order to improve the performance of our optimization, we have exploited some key observations of our problem: First, each actuator drives the motion on a 1D curve. This means that instead of considering the 3D displacement of vertices, we can only consider their distance from the neutral pose. Second, target motion that resides within the avatar's range is reproduced fairly well, while large motion is clamped. Considering the relative position (the ratio of every vertex's distance from the

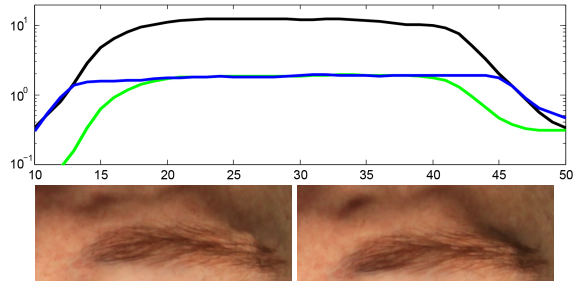


Figure 4: *Temporal remapping. Top: The graph shows the displacement of a vertex on the edge of an eyebrow. The original motion (black) surpasses the avatar's motion gamut. Static physical simulation matches the motion only within the gamut (blue). The remapped motion matches dynamic behavior instead (green). Bottom: The eyebrow position at frame 43. The projected features are nearly nonexistent while the eyebrow in the original motion still stays at peak position (left).*

neutral pose to its maximum distance in the performance) allows a description of the motion relative to gamut of the target as well as actuated performance. Incorporating all these observations, we finally get the following error term for a vertex v in a target performance mesh $\mathcal{T}_n$ and its corresponding position u in an actuated one $\hat{\mathcal{M}}_n$:

$$d(v,u) = |\vec{U}|\left(\frac{1}{|\vec{V}|}\frac{\partial|\vec{v}|}{\partial t} - \frac{1}{|\vec{U}|}\frac{\partial|\vec{u}|}{\partial t}\right)\cdot\omega_g + |\vec{U}|\left(\frac{|\vec{v}|}{|\vec{V}|} - \frac{|\vec{u}|}{|\vec{U}|}\right)\cdot\omega_s, \quad (1)$$

where $\vec{v}$ is the displacement of v from the neutral pose in the aforementioned frame, $\vec{V}$ is the maximum displacement of v in the whole sequence, and $\vec{u}$ and $\vec{U}$ are their counterparts in the actuated motion. We observed that adding the relative position error term prevents the solution for converging to a local minima. In our experiments we used the values of 0.85 and 0.15 for ω_g and ω_s, respectively.

Solution procedure. The optimization process starts with the initial guess that reproduces the original actuated motion $\tau = (1, 2, ..., N)$. During the optimization process, given the vector τ, we generate the induced actuated mesh sequence $\hat{\mathcal{M}}_n, n = 1..N$, and compute the aforementioned error term for a pre-selected random subset of the vertices. The error function used by the optimization $d : [1..N]^N \rightarrow \mathbb{R}$ is the Frobenius norm of the matrix containing all the error measures per vertex per frame. As this function is piecewise linear, its gradient can be computed analytically for each linear segment. To prevent local minima, we iteratively perturb the solution to generate new initial guesses by randomly sampling $\tau_n = [\tau_{n-1}, \tau_{n+1}]$ until there is no improvement of the solution in the current iteration. Finally, we replay the re-timed performance with the physical avatar and scan the exact geometry of $\hat{M}_n$ to obtain pixel-accurate data.

4.3 Detail Remapping

Having the re-timed avatar geometry, the next step is to map the details of the target performance to the avatar. The task of mapping one geometry to another is an ambiguous one, as some regions should be mapped to their semantic counterparts, such as the eyebrows in our case, while other regions, such as the lips, should deform freely to enhance expressiveness (see for example Figure 10). Therefore, we propose a method that does not alter geometry, but textures the avatar. This is done by rendering the performance from several points of view, deforming the rendered images to match

the avatar according to user-specified semantics, and back-projects these images to the avatar while blending them in a confidence-driven manner.

Appearance transfer. Given a target performance sequence, consisting of N frames represented by a coherent set of meshes $\mathcal{T}_n, n = 1..N$, and a correlating sequence of the avatar $\hat{\mathcal{M}}_n$, the process starts with computing the correspondence between the neutral pose of the target performance, denoted by $\mathcal{T}_0$, and the neutral pose of the avatar $\mathcal{N}$. Using the method described in Section 4.1, the correspondence is achieved by registering $\mathcal{T}_0$ onto $\mathcal{N}$. Next, for every frame $\mathcal{T}_n$, we render it from m viewpoints, where $m = 4$ in our case. We carefully picked the views such that the complete facial area is covered. The result is a set of images $\mathbf{I}_i^{T_n}, i = 1..m$ and corresponding depth maps $\mathbf{Z}_i^{T_n}, i = 1..m$. As the avatar's meshes potentially cover more of the avatar itself than the target performance, we expand the target information of the rendered images $\mathbf{I}_i^{T_n}$ by mirroring the image across the mesh boundaries, adding a blurring term that grows with the distance from the boundary. While we achieved satisfactory results, in theory more sophisticated hole filling or texture generation algorithms could be used. Boundaries are determined by transitions between background and non-background depths in the depth maps $\mathbf{Z}_i^{T_n}$. The avatar's corresponding frame is also rendered, after being rigidly aligned with $\mathcal{T}_n$, creating the $\mathbf{I}_i^{\hat{M}_n}$ and $\mathbf{Z}_i^{\hat{M}_n}$ counterparts. Next, we deform the images $\mathbf{I}_i^{T_n}$ to match their avatar's counterparts, using moving least squares [Schaefer et al. 2006]. The deformation is driven by a subset of vertices, which constrain the pixels they are projected to in $\mathbf{I}_i^{T_n}$ to move the projected position of their corresponding vertices in the avatar's rendering. Implicitly, this process deforms the low-frequency behavior of the target performance to match the avatar's one, while keeping true the high-frequency behavior of the target performance. The choice of the driving vertices is elaborated upon later in this section. Next, the images are projected back onto $\hat{\mathcal{M}}_n$, which means that every vertex receives the color from its rendered position on the deformed images, if it is not occluded. Blending between the different viewpoints is done based on the confidence of the vertex's color, determined by the cosine of the angle between the surface normal and viewing direction. As a final step, we perform for every vertex a few Laplacian temporal smoothing iterations on the resulting colors.

Conveying semantics. As aforementioned, the target performance is rendered and the images are deformed to match the physical avatar. The goal of the deformation is to adapt the target's features to the avatar while preserving the artistic intent of the performance. This notion suggests different behavior for different animations, and we allow the user to indicate the semantics of the animation by selecting individual or curves of vertices of the target performance and assign a property to it. These properties affect the behavior of the image deformation step described before. We have found that dividing the vertices into three types was sufficient to convey the semantics in our examples, and have used the same categorization for all of them. By default, all vertices are categorized as *free to move*, and have no effect on the image deformations. The second type, marked as *geometrical constraint*, enables the user to define vertices that will constrain the pixels that they are rendered to. The corresponding pixels of these vertices are moved to the position that their avatar's counterpart was rendered to, given that both are not occluded in the images. This type of constraint is usually used for vertices which are static throughout the performance, such as the nose, and is also useful for regions that should accurately match, such as the edges of the mouth and the eyebrows. The last type, marked as *view-dependent constraint*, relates to the fact that the geometries of the target performance and the avatar head do not match perfectly in some regions, and therefore the projection differs depending on the point of view. Marking these types of vertices with an associated viewpoint means that these vertices are constrained to match the avatar vertices they were projected closest to during the marked viewpoint. Figure 5 illustrates the effect of the different types of constraints: Removing the geometrical constraint from the eyebrows results in their projection on the middle of the forehead. Additionally, vertices that are marked as constrained with a front point of view are changed to a side one. This change proves unnatural from the front when the lip deviates away from the animatronic's geometry. In all our experiments, we have used 8 curves and 20 individual vertices that were geometrically constrained, and 2 curves and 5 individual vertices that were constrained view-dependent from the front view. Note that we have also experimented with different effect radii and also other types of constraints, such as snapping vertices back if they left the avatar's silhouette, but eventually found them unnecessary for our application.

Figure 5: *Semantics illustration. The marked vertices (left) are of the geometric type (red) and view-dependent type (blue). Removing the constraints off the eyebrows results in an unnatural positioning (2^{nd} and 3^{rd} images). Vertices on the lips, marked with front view-dependent constraint, are changed to a side one, which yields an unnatural look from the front (4^{th} and 5^{th} images).*

5 Projection

After preprocessing the geometry and finally generating the per-vertex colors containing the desired shading, the model has to be projected accurately onto the physical avatar. This step involves rendering the geometry from the calibrated projector views and distorting the images to compensate for lens distortion. Additionally, we compute a light transport matrix that is used in a global optimization step for blending of multiple projector contributions, neutralizing physical light drop-off effects, and compensating for defocus and subsurface scattering to generate an optimized reproduction of high frequencies. To achieve this goal, besides the geometric calibration already described in Section 4, further data acquisition steps have to be carried out. Therefore the response curves of the used devices were linearized to simplify the image analysis and processing steps. While the camera response curves were linearized using the method described in [Debevec and Malik 1997], a Spyder4ELITE colorimeter was used for projector linearization and to match their color gamuts as well as lumen output. To match the cameras' color gamuts, an x-rite ColorChecker Classic based color transformation calibration was carried out.

5.1 Defocus Data Acquisition

To accurately compensate for the projection defocus, the used PSF has to represent the physical defocus as precisely as possible. Following [Nagase et al. 2011; Aliaga et al. 2012], we approximate the projector defocus by a two-dimensional isotropic Gaussian function in the projector's image coordinate, depending on the pixel position

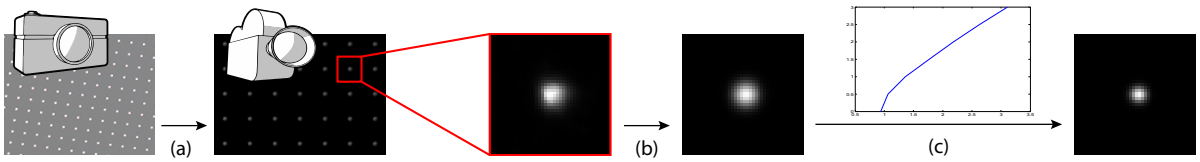


Figure 6: *Overview of the defocus measurement pipeline.* (a) *Back projection of the captured images to the projector's image plane and normalization.* (b) *Gauss fitting for each captured blob.* (c) *Recovering the amount of projector blur from the precomputed* LUT.

and the distance to the projector:

$$PSF_z(xy, xy') = e^{-\frac{(x-x')^2+(y-y')^2}{\sigma^2_{x,y,z}}}.$$

Here, x and y are pixel coordinates of the pixel from which the projected light originates, x' and y' are the pixel coordinates of the target pixel that is illuminated by the defocused pixel, and z is the distance to the projector in world coordinates of the surface corresponding to the target pixel. σ is the standard deviation of the Gaussian function.

The PSF measurement process is based upon the one proposed in [Nagase et al. 2011]. The projector displays a two-dimensional grid of white pixels on black background onto a white, planar surface that is oriented to be orthogonal to the projection axis of the projector. This surface is placed at different distances around the focal plane of the projector and images are taken of the projected pixel pattern using one or more cameras. The Gaussian function is defined in the coordinate frame of the projector, requiring that all captured images be projected into the projector's image plane. Our implementation uses homographies [Sukthankar et al. 2001] for this purpose. Each back-projected image is split into patches, one for each projected pixel, and the PSF model is fitted to each patch, resulting in a σ value and a position x and y for each image patch. As our projectors did not exhibit significant chromatic aberrations, we captured only white patterns. In this case, the position (x and y) can be ignored, as any deviation of those coordinates from the coordinates of the originally projected pixel can be explained by inexact back projection. Using the computed homographies in combination with the geometrically calibrated cameras and projectors, we also compute the distance to the projector for each pattern.

The σ values together with their respective distances and pixel coordinates constitute a dense, irregular field of defocus measurements, called a PSF field, that will be used to build the equation system for compensation. Depending on the density of the measurements, the defocus values for each point inside the covered volume can be interpolated with high accuracy. We observed that even while taking measures to reduce errors and minimize the influence of noise and environment light, the proposed measurement procedure produces σ values much greater than 0, even when measuring next to the focal plane. In our setup the minimal σ values were around 0.8. As our PSF model describes Gaussian functions in the projector image space, a σ value of 0.8 translates into a Gaussian that includes already severe defocus, covering multiple neighboring pixels. Reasons for this additional defocus include coma and chromatic aberrations of the camera lenses, its aperture settings, sampling inaccuracies both on the camera CCD and during the back projection step, and noise.

We propose an additional calibration step, referred to as sigma calibration, designed to uncover the blurring behavior of the capturing and model fitting pipeline. For this, we place the same white plane that was used for the measurements above into the focal plane and project a single pixel on a black background, followed by Gaussian blurred versions of the same with increasing σ. The captured patterns were again fitted to Gaussians, which results in a lookup table (LUT) between the σ values of the actually projected Gaussian functions and the ones found using the measurement pipeline. The overall process is illustrated in Figure 6.

Besides measurement of the projector defocus, subsurface scattering is measured and modeled as well. The modeling was done using the method described in [D'Eon and Irving 2011] while the measurement was carried out using a device based on [Weyrich et al. 2006].

5.2 Projection Image Computation

To optimize the projected images, the light transport is computed and compensated for. We modeled the light transport as matrix-vector multiplication:

$$C = LP, \tag{2}$$

where P is a vector containing the projected images, L is a matrix containing the light transport, and C is the output of the system. The semantic meaning of C depends on what aspect of the projection system is of interest. In previous works in the context of light transport and defocus compensation [Zhang and Nayar 2006; Aliaga et al. 2012; Wetzstein and Bimber 2007], C corresponds to an image captured by a designated camera that is used as a proxy for a human observer, and L encodes the light transport from one or more projectors to this camera.

To the best of our knowledge, we present the first work on pre-correcting defocus compensation for multi-projector systems that does not use a reference camera as optimization target. Instead, we completely work in the image planes of the involved projectors, treating them as virtual cameras. In this case, C represents the set of images that would be captured by the projectors. As the compensation images are generated using the parameters stored directly for each projector pixel, the resulting compensation is independent of the camera viewpoint and thus is not influenced by occlusions, obliqueness, camera defocus, etc., which would occur from almost any camera viewing position.

Compensation of the light transport, i.e. finding the images P that produce the output C when being projected, conceptually involves an inversion of the light transport: $P' = L^{-1}C'$. Here C' is the desired output of the system and P' is the input that produces it when projected. In most cases, directly inverting L is impossible because L is not full rank. As was done in [Zhang and Nayar 2006] and [Aliaga et al. 2012], we instead reformulate the compensation as a minimization problem: $P' = \text{argmin}_{0\leq P\leq 1}\|LP - C'\|^2$. In the course of building up the components of the equation system, this minimization will be extended to contain locally varying upper bounds, weighting of individual pixels, and additional smoothness constraints, resulting in the following minimization:

$$\begin{aligned} P' &= \underset{0\leq P\leq U}{\text{argmin}} \|W\,(TP - S)\|^2 \\ &= \underset{0\leq P\leq U}{\text{argmin}} \|W\left(\begin{bmatrix} L \\ Smooth \end{bmatrix} P - \begin{bmatrix} C \\ 0 \end{bmatrix}\right)\|^2. \end{aligned} \tag{3}$$

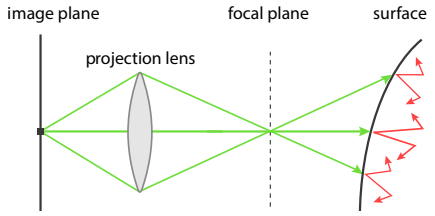


Figure 7: *Simplified visualization of the spatial distinction between projector defocus resulting from its lens properties (green) and subsurface scattering (red). The defocus originates before the light physically reaches the surface, while subsurface-scattering evolves only once it has hit the surface.*

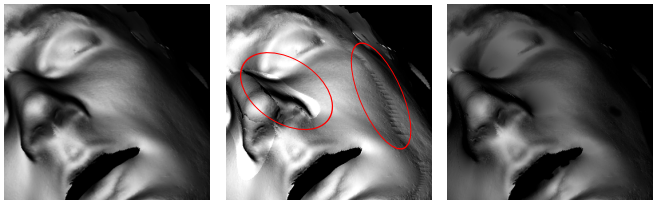
S is a vector containing the target images C' and the smoothing target values of constant 0. T is a matrix consisting of the light transport L and the smoothing terms $Smooth$. W is a diagonal matrix containing weights for each equation. Finally, U contains the upper bounds of the projected image pixel values.

Light Transport. Below, we build up the light transport iteratively by its components. For projector defocus, σ is looked up in the PSF field at the pixel coordinates of the source pixel as well as at the depth of the target pixel. The PSF model is then evaluated using σ, and the resulting value is normalized such that all the light emitted at the same source pixel sums up to 1. To ensure that the compensated pictures result in a uniformly bright appearance, light drop-off caused by distance to the projector and the incidence angle of the light at the surface is included in the light transport. This is done by multiplying the light drop-off factor on top of the defocused projection computed previously.

As illustrated in Figure 7, subsurface scattering physically happens after projector defocus. Thus it is possible that light emitted from one pixel can travel to the same target pixel using multiple paths, so care has to be taken to sum up those contributions correctly. The subsurface scattering factor is looked up in the previously measured scattering profile with the world coordinate distance between the two involved surface points. This formulation is not quite correct, as these measurements are valid only for flat patches of silicone with a certain thickness. General surfaces are neither of uniform thickness nor flat, however, and especially in concave parts, the point distance in world coordinates does not correspond to the distance on the surface. But these inaccuracies are relatively small and don't carry much weight when compared to other sources of errors, such as inexact geometry and calibrations. As such, we do not handle these effects.

This finishes the single projector light transport (PLT). The following modifications are needed only in multi-projector systems; they fill in the cross PLT without changing the already computed values. Instead of recomputing projector defocus and subsurface scattering for the cross PLT, the relevant values are looked up in the results of the single PLT using a projective mapping between the projectors. See the appendix for a description of this lookup process. To make sure that the computed cross PLT actually deals in correct units, the relative brightness of the involved projectors has to be considered as well. We use three projectors of the same make and model, and have calibrated them to be of the same brightness as part of the projector response curve linearization mentioned earlier.

Blending Multiple Contributions. In multi-projection systems, blending maps are applied to ensure consistent intensities in overlapping projection areas (cf. e.g. [Raskar et al. 1998; Harville et al. 2006]). This is especially important when projecting onto objects



(a) (b) (c)

Figure 8: *Blending Comparison.* (a) *Input image.* (b) *Compensation image without blending; note the marked artifacts.* (c) *Compensation image with blending maps as upper bounds.*

that are discontinuous when seen from a specific projector. We use a geometry-based blending map calculation approach using shadow volumes to detect discontinuous regions in the projector image planes and smoothly fade out the individual projector intensities in these areas as well as at the edges of the image planes in overlapping areas.

Previous work on multi-projector defocus compensation, such as [Aliaga et al. 2012], does not take blending into account. This can be a serious shortcoming, as it produces noticeable artifacts in the presence of discontinuities. Not involving blending maps while at the same time compensating for light drop-off caused by incidence angle has the effect that projectors increase their intensity when projecting onto oblique surfaces, instead of leaving the illumination of such surfaces to another projector in a more suitable position.

We propose to include the blending maps into the minimization as upper bounds (U in equation 3). See Figure 8 for a comparison of compensation results with and without blending. These results were computed for a three-projector system (see Figure 3), and the compensation images of the lower projector are shown. It can be seen that the result without blending (b) contains severe artifacts. They are most noticeable in areas of discontinuities such as around the nose and on the cheeks. Applying the proposed approach reduces the artifacts below a perceptual level (c). In regions where projectors overlap, one point on the target surface is represented by multiple pixels in the image planes of multiple projectors. If each of those pixels had the same weighting in the residual computation, overlapping regions would be treated as more important than non-overlapping regions. Not all solution pixels have the same accuracy requirements: It is more important for each projector to find good solutions for image patches for which it is the only projector, or onto which it projects orthogonally. These criteria are also followed when constructing blending maps, which makes blending maps good weights for the individual equations in the system (W

(a) (b) (c)

Figure 9: *Weighting comparison. (Close-up of the image shown in Figure 8)* (a) *Input image, showing the nose.* (b) *Compensation image with blending maps as upper bounds but no weighting, leading to artifacts (red).* (c) *Adding the blending maps as weights removes those errors.*

Figure 11: *Results of a single frame captured from random viewing angles, illuminated only by projectors (top row) and with ambient lighting in the room (bottom row).*

in equation 3). See Figure 9 for a comparison of compensation results with and without weighting. These images show an excerpt around the nose of the same three-projector system as before. (b) was computed with blending maps as upper bounds, but without weighting the equations. Note the artifacts (in red) that disappear when including the weights, resulting in (c).

Smoothing. Even careful PSF measurement and sigma calibration might lead to a slight under- or overestimation of the projector defocus, resulting in visible artifacts caused by the projection of incorrect compensation images. Additionally, in regions where multiple projectors overlap, there is no guarantee in which way the compensation image is composed. This can lead to the case that for two neighboring pixels, one pixel is completely produced by the first projector and the other by the second projector. In this case, small calibration errors will become immediately apparent. Both of these issues can be reduced by introducing additional smoothness constraints. We implemented smoothness constraints similar to the ones proposed in [Aliaga et al. 2012]. We refer to the supplemental material for a description of the smoothness constraints.

Solving. In our implementation, we used the iterative, constrained, steepest descent algorithm presented in [Zhang and Nayar 2006] as a solver for the equation system. See the supplemental material for a description of how to deal with the global scaling of the system.

6 Results and Discussion

To evaluate the performance of our projection-based enhancements, we used the silicone animatronic head described by Bickel et al. [2012] and mapped a performance capture sequence of a real actor onto it. In addition, some of the input sequences were artistically altered to simulate a man older than the one who actually performed. Figure 10 shows different results of our proposed method.

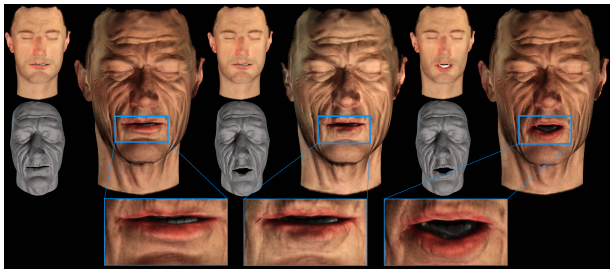


Figure 12: *The neutral pose (left) of a sequence, compared to an extreme pose while the animatronic head is kept static (middle), and while it is actuated according to the proposed method (right). The head configuration under uniform illumination, and the target appearances are shown to the left of each result.*

As can be seen, the actuators of the animatronic are not able to generate the complex skin deformation required to accurately reproduce the input geometry. Adding the missing information using our proposed projection mapping significantly enhances the high-frequency components, and thus the expressiveness of the performance. As the process is designed to optimize for several viewing angles, we demonstrate the robustness to viewer positions by using a hand-held camera, with and without ambient lighting in the room, in the accompanying video and in Figure 11. Furthermore, to emphasize the effect of our two-scale approach, we keep the animatronic head static, and perform the suggested method for one of the sequences. As can be seen in the accompanying video, as well as in Figure 12, while an illusion of the desired performance can be generated using only the projection or only physical animation, the combination of the two produces a far more compelling result.

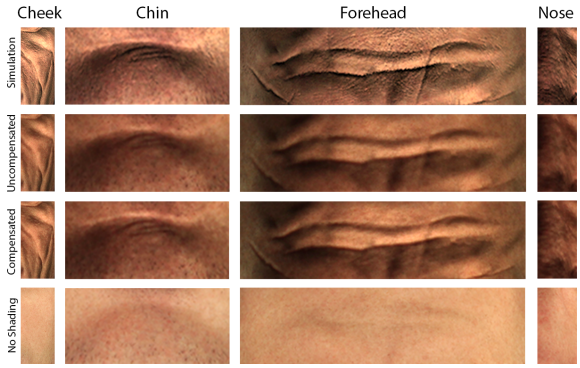


Figure 13: *Close-up comparison of the projection shown in Figure 1. Upper row: Simulations of the desired appearance. Second row: Uncompensated results. Third row: the compensated results. Fourth row: Appearance of the head when no shading is projected.*

To evaluate the quality improvement of our multi-projector optimization method, we used the structural similarity index (SSIM) [Wang et al. 2004], which is a method for assessing the perceptual quality of a distorted image when compared to the original. We used a modified version of SSIM to compare the projection results of uncompensated and compensated shadings to a ground truth image. This was generated by rendering the input image from a calibrated camera and using the color mapping technique described in [Grundhöfer 2013] in an inverse manner to simulate the per-pixel surface color modulations. It has been modified in that it does not take the absolute pixel values into account, but only compares contrast and structure. This measure results in a value between -1 and 1 where 1 corresponds to no distortion.

Figure 10: *Three captured results of extreme poses, generated with our system. For every frame, the desired appearance is on the bottom left, and the robot configuration, under uniform white illumination, is on the top left.*

Figure 13 shows the excerpts from the final frame of the growing-old sequence shown in Figure 1. The uncompensated and the compensated projections were compared to the simulated ground truth, resulting in the SSIM scores contained in Table 1. As can be seen, besides the perceived improvements presented in Figure 13, the defocus compensation results in a measurable increase in similarity to the simulated ground truth. As the finite pixel resolution as well as the subsurface scattering properties of the silicone skin constrain the reproduction quality of the input shading, a value of 1.0 is impossible to achieve with the presented setup. In practice we found compensating for subsurface scattering to be more important than for defocus. Experiments showed that for the silicon head, the subsurface scattering compensation enhanced the image quality significantly more when compared to defocus compensation alone, as it also reduces image contrast in well-focused areas.

On our machine featuring a quad-core i7 Intel CPU, 24 GB of RAM, and an NVidia QuadroPlex graphics card, the creation of a detailed mesh out of the acquired point cloud lasts about 3 minutes per frame, and the application of a target frame onto it takes about 1 minute. The actual projection image generation is performed in real-time. The computation of a light transport matrix for one pose takes about 11 minutes, while the compensation of a set of projection images takes about 8 minutes.

7 Summary and Future Work

In this paper we presented a novel approach using spatially varying illumination to enhance the appearance and expressiveness of a silicone-skin-based head animatronic. We demonstrated that a carefully calibrated multi-projector system in combination with geometrical mapping can significantly enhance its realism by projecting high-frequency skin structures that cannot be reproduced by the animatronic's actuators and the silicone skin alone.

In the future, we are planning to integrate the proposed approach into a real-time, live feedback system to enable a realistic and responsive animatronic interaction. While the software tools for real-time geometry mapping are already available, this step requires a sophisticated engineering effort in terms of accurate hardware setup and synchronization. The used subsurface scattering compensation uses a simplified, spatially uniform description of the subsurface scattering behavior. While this is the result of a missing measurement device, a future research direction would be the utilization of the projector-camera system to acquire accurate, spatially varying subsurface scattering information similar to the work presented in [Ghosh and Debevec 2008]. Another related future research direction would be an accurate estimation of the spatially varying surface BRDF to also enable a view-independent photometric projector compensation.

Area	No Shading	Uncompensated	Compensated
Cheek	0.543	0.816	0.864
Chin	0.722	0.889	0.908
Forehead	0.632	0.815	0.849
Nose	0.669	0.864	0.885

Table 1: *SSIM evaluation results for the cropped image regions shown in Figure 13.*

Acknowledgments

We thank our actor, Roland Angst as well as Craig Gotsman and the anonymous reviewers for their helpful comments. We would also like to express our sincere gratitude to Thabo Beeler and Derek Bradley for insightful discussions and providing the performance capture data.

References

ALIAGA, D. G., YEUNG, Y. H., LAW, A., SAJADI, B., AND MAJUMDER, A. 2012. Fast high-resolution appearance editing using superimposed projections. *ACM Trans. Graph. 31*, 2, 13:1–13:13.

BEELER, T., HAHN, F., BRADLEY, D., BICKEL, B., BEARDSLEY, P., GOTSMAN, C., SUMNER, R. W., AND GROSS, M. 2011. High-quality passive facial performance capture using anchor frames. In *ACM Trans. Graph.*, vol. 30, ACM, 75.

BICKEL, B., KAUFMANN, P., SKOURAS, M., THOMASZEWSKI, B., BRADLEY, D., BEELER, T., JACKSON, P., MARSCHNER, S., MATUSIK, W., AND GROSS, M. 2012. Physical face cloning. *ACM Trans. Graph. 31*, 4.

BIMBER, O., AND EMMERLING, A. 2006. Multifocal projection: A multiprojector technique for increasing focal depth. *Trans. Visualization and Computer Graphics 12*, 4, 658–667.

BIMBER, O., IWAI, D., WETZSTEIN, G., AND GRUNDHÖFER, A. 2007. The Visual Computing of Projector-Camera Systems. In *Proc. Eurographics (State-of-the-Art Report)*, 23–46.

BLANZ, V., AND VETTER, T. 1999. A morphable model for the synthesis of 3d faces. In *Proc. Computer graphics and interactive techniques*, 187–194.

CHUANG, E., AND BREGLER, C. 2002. Performance driven facial animation using blendshape interpolation. *Computer Science Technical Report, Stanford University 2*, 2, 3.

DEBEVEC, P. E., AND MALIK, J. 1997. Recovering high dynamic range radiance maps from photographs. In *Proc. of ACM SIGGRAPH*, 369–378.

D'EON, E., AND IRVING, G. 2011. A quantized-diffusion model for rendering translucent materials. In *ACM Trans. Graph.*, vol. 30, 56.

EKMAN, P., AND FRIESEN, W. V. 1977. Facial action coding system.

GHOSH, A., AND DEBEVEC, P. 2008. Estimating multi-layer scattering in faces using direct-indirect separation. In *ACM SIGGRAPH 2008 talks*, SIGGRAPH '08, 2:1–2:1.

GROSSE, M., WETZSTEIN, G., GRUNDHÖFER, A., AND BIMBER, O. 2010. Coded aperture projection. *ACM Trans. Graph. 29*, 3, 22:1–22:12.

GRUNDHÖFER, A. 2013. Practical non-linear photometric projector compensation. In *2nd Int. Workshop on Computational Cameras and Display*.

HARTLEY, R. I., AND ZISSERMAN, A. 2004. *Multiple View Geometry in Computer Vision*. Cambridge University Press.

HARVILLE, M., CULBERTSON, B., SOBEL, I., GELB, D., FITZHUGH, A., AND TANGUAY, D. 2006. Practical methods for geometric and photometric correction of tiled projector. In *Computer Vision and Pattern Recognition Workshop*, 5–5.

HAVALDAR, P., PIGHIN, F., AND LEWIS, J. 2006. Performance driven facial animation. In *ACM SIGGRAPH Courses*.

ISHIGURO, H. 2006. Interactive humanoids and androids as ideal interfaces for humans. In *Proc. International Conference on Intelligent user interfaces*, 2–9.

KAZHDAN, M., BOLITHO, M., AND HOPPE, H. 2006. Poisson surface reconstruction. In *Proc. SGP*.

KURATATE, T., MATSUSAKA, Y., PIERCE, B., AND CHENG, G. 2011. Mask-bot: A life-size robot head using talking head animation for human-robot communication. In *Int. Conference on Humanoid Robots (Humanoids)*, 99 –104.

LAW, A. J., ALIAGA, D. G., SAJADI, B., MAJUMDER, A., AND PIZLO, Z. 2011. Perceptually based appearance modification for compliant appearance editing. *Comput. Graph. Forum 30*, 8, 2288–2300.

LI, H., SUMNER, R. W., AND PAULY, M. 2008. Global correspondence optimization for non-rigid registration of depth scans. In *Computer Graphics Forum*, vol. 27, 1421–1430.

LI, H., WEISE, T., AND PAULY, M. 2010. Example-based facial rigging. *ACM Trans. Graph. 29*, 4, 32.

LINCOLN, P., WELCH, G., NASHEL, A., ILIE, A., STATE, A., AND FUCHS, H. 2009. Animatronic shader lamps avatars. In *Proc. Int. Symposium on Mixed and Augmented Reality*, 27–33.

LIPMAN, Y., SORKINE, O., LEVIN, D., AND COHEN-OR, D. 2005. Linear rotation-invariant coordinates for meshes. *ACM Trans. Graph. 24*, 3, 479–487.

MISAWA, K., ISHIGURO, Y., AND REKIMOTO, J. 2012. Ma petite cherie: what are you looking at?: a small telepresence system to support remote collaborative work for intimate communication. In *Proc. Augmented Human International Conference*, ACM, New York, NY, USA, AH '12, 17:1–17:5.

MOUBAYED, S. A., EDLUND, J., AND BESKOW, J. 2012. Taming mona lisa: Communicating gaze faithfully in 2d and 3d facial projections. *ACM Trans. Interact. Intell. Syst. 1*, 2 (Jan.).

NAGASE, M., IWAI, D., AND SATO, K. 2011. Dynamic defocus and occlusion compensation of projected imagery by model-based optimal projector selection in multi-projection environment. *Virtual Real. 15*, 2-3 (June), 119–132.

NISHIO, S., ISHIGURO, H., AND HAGITA, N. 2007. *Humanoid Robots: New Developments*. I-Tech, ch. Geminoid: Teleoperated Android of an Existing Person.

NOH, J.-Y., AND NEUMANN, U. 2001. Expression cloning. In *Proc. Conf. on Comp. Graph. and Int. Techniques*, 277–288.

OYAMADA, Y., AND SAITO, H. 2008. Defocus blur correcting projector-camera system. In *Proc. Int. Conference on Advanced Concepts for Intelligent Vision Systems*, 453–464.

RASKAR, R., WELCH, G., CUTTS, M., LAKE, A., STESIN, L., AND FUCHS, H. 1998. The office of the future: A unified approach to image-based modeling and spatially immersive displays. In *Proc. Conf. on Comp. Graph. and Int. Techniques*, 179–188.

RASKAR, R., WELCH, G., LOW, K.-L., AND BANDYOPADHYAY, D. 2001. Shader lamps: Animating real objects with image-based illumination. In *Proc. Eurographics Workshop on Rendering Techniques*, 89–102.

SCHAEFER, S., MCPHAIL, T., AND WARREN, J. 2006. Image deformation using moving least squares. In *ACM Trans. Graph.*, vol. 25, 533–540.

SEN, P., CHEN, B., GARG, G., MARSCHNER, S. R., HOROWITZ, M., LEVOY, M., AND LENSCH, H. 2005. Dual photography. *ACM Trans. Graph. 24*, 3, 745–755.

SEOL, Y., LEWIS, J., SEO, J., CHOI, B., ANJYO, K., AND NOH, J. 2012. Spacetime expression cloning for blendshapes. *ACM Trans. Graph. 31*, 2, 14.

SUKTHANKAR, R., STOCKTON, R. G., AND MULLIN, M. D. 2001. Smarter presentations: Exploiting homography in camera-projector systems. In *Proc. Int. Conference on Computer Vision*, vol. 1, IEEE, 247–253.

TENA, J. R., HAMOUZ, M., HILTON, A., AND ILLINGWORTH, J. 2006. A validated method for dense non-rigid 3d face registration. In *Int. Conf. on Video and Signal Based Surveillance*.

WANG, Z., BOVIK, A. C., SHEIKH, H. R., AND SIMONCELLI, E. P. 2004. Image quality assessment: From error visibility to structural similarity. *Trans. on Image Processing 13*, 4, 600–612.

WETZSTEIN, G., AND BIMBER, O. 2007. Radiometric compensation through inverse light transport. In *Computer Graphics and Applications, 2007. PG '07.*, 391 –399.

WEYRICH, T., MATUSIK, W., PFISTER, H., BICKEL, B., DONNER, C., TU, C., MCANDLESS, J., LEE, J., NGAN, A., JENSEN, H. W., ET AL. 2006. Analysis of human faces using a measurement-based skin reflectance model. *ACM Trans. Graph. 25*, 3, 1013–1024.

ZHANG, L., AND NAYAR, S. 2006. Projection defocus analysis for scene capture and image display. *ACM Trans. Graph. 25*, 3, 907–915.

ZHANG, Z. 2000. A flexible new technique for camera calibration. *Trans. Pattern Anal. Mach. Intell. 22*, 11 (Nov.), 1330–1334.