

Light-Field Appearance Editing Based on Intrinsic Decomposition

Shida Beigpour, Sumit Shekhar, Mohsen Mansouryar, Karol Myszkowski, and Hans-Peter Seidel

Abstract—We present a framework for image-based surface appearance editing for light-field data. Our framework improves over the state-of-the-art without the need for a full “inverse-rendering”, so that full geometrical data, or presence of highly specular or reflective surfaces are not required. It is robust to noisy or missing data, and handles many types of camera array setup ranging from a dense light-field to a wide baseline stereo-image pair. We start by extracting intrinsic layers from the light-field image set maintaining consistency between views. It is followed by decomposing each layer separately into frequency bands, and applying a wide range of “band-sifting” operations. The above approach enables a rich variety of perceptually plausible surface finishing and materials, achieving novel effects like translucency. Our GPU based implementation allow interactive editing of an arbitrary light-field view, which can then be consistently propagated to the rest of the views. We provide extensive evaluation of our framework on various datasets and against state-of-the-art solutions.

Index Terms—Light-Field, Material Perception, Multi-View, Stereo, Filtering, Image-based Rendering

I. INTRODUCTION

Light-field technology offers many advantages with respect to traditional 2D imaging, as it enables depth estimation, refocusing, as well as view-dependent effects such as glossy reflections and motion parallax that are desirable in many applications such as virtual reality (VR). Typically narrow baseline light-fields with dense angular views and low spatial resolutions are considered due to the accessibility of inexpensive capturing hardware such as the Lytro camera. Since such light-fields exhibit strong redundancy of data between views and offer only a limited freedom in virtual camera placement and manipulation, sparse and wide baseline light-fields are increasingly gaining attention [1]. In this work, we present a framework to extract consistent intrinsic components (*e.g.*, shading and reflectance) of sparse light-field data in order to simulate different perceptual effects that alter the appearance of the objects in the scene. Unlike most light-field based methods [2], [3], we do not make any strong assumptions on the structure of the light field data. We do not require high number of views or small baselines, however our approach generalizes to such cases as well.

Surface appearance and material editing is often achieved by a full inverse rendering where a 3D geometry is estimated along with an environment map [4], and then an altered version of the scene is rendered. The quality of such results are subject to the soundness of the extracted geometry and environment map, which might require highly specular objects in the scene

[4], [5] or the sky visibility [6]. Even modest amounts of inaccuracy or noise in the reconstructed 3D model and lighting can lead to visible artifacts that might easily ruin all material editing efforts. In fact, such inverse rendering approaches might not be strictly required, as recent findings on material discrimination and recognition indicate that the human visual system (HVS) does not perform physically correct inverse optics simulation [7], [8].

A general study of such heuristics is done in [9], [10]. Specific analysis of human perception with respect to *glossiness* and *translucency* is carried out in [8] and [7], respectively. Further in [11] the authors discuss the link between spatial frequency bands of an image and material perception. Overall it has been shown that HVS relies on built-in heuristics that connect certain image patterns with material properties. The above indicates that editing image patterns by skillful filtering of different intrinsic layers may provide better visual quality than artifacts-prone full inverse rendering.

Inspired by the *band-sifting* concept, proposed by Boyadzhiev *et al.* [12], we simulate the appearance of different materials and surface structures consistently on light-field data. Instead of band-sifting the luminance channel, we process intrinsic layers, which significantly increases the variety of material edits, as they can precisely be targeted on textures, geometric details, or glossiness. Due to intrinsic layer separation we can introduce new, perceptually-justified band-sifting operations, which lead to meaningful appearance changes such as opaque-to-translucent object conversion. We can also make more profound editing effects, for the appearance changes that have already been demonstrated by Boyadzhiev *et al.*, without unwanted side-effects.

Contributions: We propose a framework for light-field appearance editing with the following contributions:

- An intrinsic image decomposition method which, unlike existing work, is capable of handling wide-baseline light-field assuring consistency between views.
- Extension of the band-sift filtering using intrinsic image layers to improve its performance and robustness while maintaining consistency between views.
- Demonstrate how to reproduce and manipulate complex perceptual appearance effects (*e.g.*, translucency, pearlecency, wetness) by means of purely image-based methods.
- GPU based interactive image editing framework.

II. RELATED WORK

In this section, we discuss existing solutions to image, video, and light-field decomposition into intrinsic components.

Then, we summarize the work on light-field manipulation, with special focus on material and appearance changes. Since such efforts are relatively sparse, we broaden our discussion to manipulation of single images and videos with similar goals in mind.

A. Intrinsic Image Decomposition

The term *intrinsic images* is first introduced in the literature by Barrow and Tenenbaum [13] to refer to mid-level components like *reflectance* and *shading*. A comprehensive survey of Intrinsic Image Decomposition methods is presented by Bonneel *et al.* [14] where different methods are categorized based on their assumptions and choice of priors. Here, we focus on methods that employ additional scene information beyond a single RGB image and discuss representative examples of such methods.

Chen *et al.* [15] use *RGB-D* to implement surface normal priors. Nonetheless, a depth image does not provide enough information about the scene to perform a full inverse rendering and complexities like *cast shadows* remain ambiguous. Similarly *stereo* matching based methods [16] use disparity to infer some level of geometric information in order to introduce additional constraints and improve the results while maintaining consistency between the two views. *Image sequences* containing camera or scene motion could be used to further resolve ambiguities [17]. *Multi-view stereo* based methods [6], [18] use a 360° view of the scene to extract full geometry and environment map.

Light-field as a special case of multi-view stereo, provides some level of angular information which could improve decomposition in the presence of view-dependent components such as specularities [3]. In addition, disparity information can be extracted which allows the inclusion of geometry prior. Similar to the case of stereo and RGB-D, geometry cues might not be helpful in resolving complexities like cast shadows. However, multiple instances of the same data might improve the decomposition robustness, *e.g.*, in the presence of noise.

To the best of our knowledge, there are two intrinsic image decomposition methods for light-fields in the literature [2], [3]. They both require highly dense light-field with small baselines. On the contrary our method is capable of handling a sparse set of views and a large baseline (*e.g.*, even a stereo pair with 20 cm baseline Fig. 13). In terms of optimization priors, we show that by focusing on reliable and well-crafted priors, we are able to outperform existing methods which often use more complex priors [14]. Our method is capable of handling both dense and sparse light-field.

Artusi *et al.* [19] provide an extensive survey on *specularity removal* in natural images. While a vast majority of intrinsic image decomposition methods ignore specularities [2], [16], [17], [20], some successful attempts have also been reported [3], [21], [22]. However, these methods focus on narrow-baseline, dense light-fields, and are not directly applicable to sparse light-fields. Also, while aiming at physically-correct specularities extraction many failure cases are reported for such methods [23] that might lead to visually disturbing artifacts, *e.g.*, for large area highlights. In our application the extraction

of specularities layer can lead to many interesting appearance editing effects. We resort to more approximate solutions that do not lead to obvious visual artifacts in our appearance editing at the expense of physical correctness. A detailed review of gloss perception is done in recent study by Chadwick *et al.* [24]. We take inspiration from [8] and [10] to extract a *psuedo* specular/highlight layer from a given image.

B. Image/Light-field Editing and Enhancement

Multi-scale edge-preserving image decompositions have been used to enhance image detail or to achieve other image appearance changes or transfers [25]–[29]. We also perform a multi-scale image decomposition, but we apply it to intrinsic image layers, which gives us a better control over the wide range of effects we can produce. Recently, deep machine learning methods have been successfully used for overall image appearance changes and various stylizations [30], but spatially selective, continuous range and large scope material editing that is intuitive to the user has not been demonstrated so far.

Light field manipulations [31] focused mostly on retargeting [32], shape deformation [33], in-painting and recolorization [34], compositing [35], and morphing [36]. However, research on light-field appearance editing is relatively sparse [37], [38], and the key problem that is considered there is intuitive propagation of sparse user edits to all views [39], [40].

C. Image/Light-field Material Editing

Material appearance editing based on single image has widely been investigated [4], [5], [41]–[45]. In all these cases, an attempt of inverse rendering has been performed. Since the problem of reconstructing all needed data from a single image is strongly underconstrained [46], manual intervention might be required to reconstruct the missing scene lighting or geometry information [5], [44]. Gryaditskaya *et al.* [23] recover such information from light-fields, which is then employed in a spatio-angular filter that enhances the roughness of glossy objects. To our knowledge, this is the only work where material appearance editing has been investigated for light-fields, and while the full inverse rendering has not been performed, the filtering quality relies on the accuracy of normal vector reconstruction. The scope of roughness manipulation is limited and even simple roughness reduction has not been shown in this framework.

Many successful editing examples have been presented in the discussed work, which was greatly facilitated by the limited sensitivity of visual system for even substantial departures from physical correctness [5], [7], [8], [47]. However, the 3D scene data, as required by these techniques, is often reconstructed with low precision, which might lead to visible artifacts, or at best reduce the generality of proposed manipulations. In this work, we rely on a much simpler, purely 2D approach, and we take an inspiration from the work of Boyadzhiev *et al.* [12] who introduce the *band-sifting* operations.

The relationship between image statistics and reflectance perception is discussed in [10], [9]. In [8] the perception of

glossiness is explored using visual cues. Band-sifting is a simple and effective technique for image based material editing based on above findings. The basic idea is to do a multiscale decomposition of luminance channel into a set of *subbands*. And then selectively manipulate these subbands based on *frequency*, *amplitude*, and *sign*. The manipulation is pretty simple where the selected part of subband is either *boosted* (multiplied by a factor greater than 1) or *reduced* (multiplied by a factor between 0 and 1). The above relationship between 2D frequency bands and material perception is discussed in [11].

We are different from [12] in the way that we introduce a new *invert* operation in addition to the previous *reduce* and *boost* operations. By applying operations individually on intrinsic layers, we are able to achieve new kinds of material editing effects. Moreover, the range of these manipulations is also increased.

Intrinsic images/videos have been used for material editing [17], [48], but only recoloring and tone mapping curve manipulations have been demonstrated respectively for the reflectance and shading layers. In this work, we additionally consider the specular layer, and a wider scope of material manipulations.

III. OVERVIEW

In this section, we provide an overview of our framework for appearance editing as shown in Fig. 1. The input light-field is first white-balanced and then decomposed into mid-level intrinsic layers (Sec. IV). These layers allow us to control different aspects of the surface appearance such as texture (reflectance), fine geometrical details (shading), and glossiness (specularity). By applying *band-sifting* operations to each layer separately and combining them together, we achieve novel appearance and material looks (Sec. V).

IV. LIGHT-FIELD INTRINSIC LAYER DECOMPOSITION

We base our intrinsic image extraction on Grosse *et al.* [49] providing a simplified *dichromatic reflection model* (Eq. 1), where an input image I is described as the sum of diffuse I^d and specular C components, and the diffuse component itself consists of shading S and reflectance R .

$$I_{\mathbf{x}} = I_{\mathbf{x}}^d + C_{\mathbf{x}} = S_{\mathbf{x}} \cdot R_{\mathbf{x}} + C_{\mathbf{x}} \quad (1)$$

Here the multiplication and addition operations are pixel-wise. For brevity, we omit pixel coordinates \mathbf{x} unless required. It is common in the literature to use logarithmic scale in order to further simplify the I^d term into $i^d = s + r$. In the above formulation lower case letters i^d , s , and r denote the respective log values of I^d , S , and R . As we are dealing with light-fields, we enforce consistency between views by jointly optimizing over *all pixels* in *all views*.

A. Pre-processing

Real-world scenes are often illuminated by non-white light sources and contain glossy surfaces with specular highlights, which is typically ignored by existing intrinsic decomposition

methods [2], [16], [17], [20]. In this work, we introduce a pre-processing step to correct illumination color and extract specularity. This allows us to simplify intrinsic image decomposition to the generic problem, where it is assumed that Lambertian surfaces are lit by achromatic light. In Fig. 9, we show our decomposition results with and without this pre-processing step for a better and fairer comparison to the above mentioned methods.

We estimate the illumination chroma using the *general grey-world* approach [50], which is based on the assumption that “the p th-Minkowski norm of a scene is achromatic after local smoothing”. We based our illumination estimation on Eq. 17 in [50], and we found that $p = 3$ performed well for our goals. To white-balance each image, we divide each pixel color by the estimation of normalized illumination color (preserving its brightness).

As our approximate specular layer extraction is based on band-sifting operations, we provide all relevant details in Sec. V-A after band-sifting is properly introduced.

B. Reflectance and Shading Layer Extraction

Current light-field decomposition methods use epipolar images to derive the intrinsic layers [2], [3]. This requires narrow baseline and dense light-field with a known structure. Our method on the other hand is designed for sparse light-fields and wide baselines where such techniques are not applicable. Therefore, we regard light-field as a generalization of stereoscopic images, and we refer to methods that consider stereo pairs [16], [51].

Bonneel *et al.* [14] states that many complex intrinsic image methods are not better than the baseline for image editing, and some priors may be even harmful. Please note that the baseline correspond to simply assigning chroma to reflectance and brightness to shading. Having an over-complex optimization with several weak (ill-posed) energy terms does not always provide a benefit. Instead, our intrinsic image optimization is tailored particularly to appearance editing needs.

Since white-balancing operation is performed in the pre-processing step (Sec. IV-A), we assume grayscale shading. The main goal in intrinsic image decomposition is decomposing pixel intensity into reflectance and shading. Many works in the literature [2], [16], [52] choose to solve for shading, and then compute reflectance using, $r = i_d - s$. We found that often existing light-field datasets contain some inconsistencies regarding image brightness due to *e.g.*, flickering light. By definition, reflectance is invariant to illumination. Therefore, we formulate our optimization by solving for reflectance instead of shading. This further simplifies formulation of our optimization:

$$\arg \min_r E(r) = \lambda_r E_r(r) + \lambda_d E_d(r) + \lambda_a E_a(r) + \lambda_s \|r\|_1 \quad (2)$$

where E_r , E_d , and E_a are retinex, disparity, and absolute shading scale terms, respectively with their corresponding weights, and λ_s is a regularization parameter assuring reflectance sparsity.

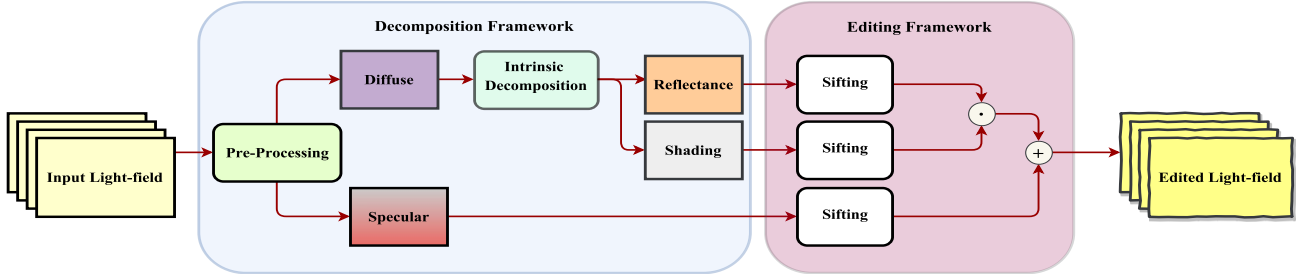


Fig. 1: Flowchart of the complete system for image-based material appearance editing in light-fields.

We use fixed weights $\lambda_r = 2$, $\lambda_d = 1$, $\lambda_a = 0.7$, and $\lambda_s = 0.1$ for all the results presented in this paper. In the supplementary materials, we provide an analysis of these parameter weights along with additional results. Below, we provide an explanation of each term.

Retinex Term: One of the most fundamental concepts in intrinsic image decomposition is *Retinex*. It can be inferred from the evaluation results in [14] that the methods which use a strong Retinex constraint achieve results better suited for image editing. Based on Retinex theory, large derivatives in the image are attributed to reflectance and small derivatives to shading. We formulate this as follows:

$$\begin{aligned}
 E_r(r) &= \sum_m \sum_{n \in N_m} [\zeta_{mn}(r_m - r_n)^2 + (s_m - s_n)^2] \\
 &= \sum_m \sum_{n \in N_m} [(1 + \zeta_{mn})(\Delta r_{mn})^2 + (\Delta i_{mn})^2 \\
 &\quad - 2\Delta i_{mn}\Delta r_{mn}] \quad (3)
 \end{aligned}$$

where m is any pixel in the light-field data, N_m are its immediate bottom and right neighbors. We used $s_m = i_m - r_m$, $\Delta r_{mn} = r_m - r_n$, and $\Delta i_{mn} = i_m - i_n$. The weight ζ_{mn} allows the optimization of reflectance to differentiate between shading and reflectance edges, smoothing out the former and preserving the latter. A crucial step for a good Retinex term is to correctly classify variations in the image (*i.e.*, edges) to shading and reflectance. Existing methods often rely only on chroma [2], [16] to find reflectance edges. This results in disregarding non-chromatic reflectance edges that are often very important (refer to numbers, eyes, and text in Fig. 10). While [2] tries to solve this issue by separately detecting the black and white pixels, both methods over-smooth the grayscale reflectance edges and are not able to preserve them (Fig. 10). Bell *et al.* [20] use a weighted variation of RGB.

We use color gradient, ∇Cg , introduced by van de Weijer *et al.* [53] to better identify true reflectance edges. This gives us sharper reflectance edges and avoids flattening and averaging of strong brightness edges in the reflectance. Many Retinex-based methods [2], [16] choose a binary weight for reflectance by applying a threshold. As we already remove the noisy values, we choose a soft threshold scheme to normalize ψ^r_{mn} in the range of $[0, 1]$. Even though color gradients perform reasonably well, we believe that a perception based reflectance edge would give better results. Modifying the edge weights based on human perception and analyzing its impact

would be an interesting idea for future work. Thus, the final weight ζ_{mn} is calculated as follows:

$$\psi^r_{mn} = \begin{cases} 0 & \nabla Cg(r_m, r_n) < 5 \\ \nabla Cg(r_m, r_n) & \text{otherwise} \end{cases} \quad (4)$$

$$\zeta_{mn} = \left(1.0 + e^{(\gamma * (\psi^r_{mn} - \alpha))}\right)^{-1} \quad (5)$$

where we found $\alpha = 5$ and $\gamma = 0.5$ to perform best in variety of scenes and datasets. The significance of the retinex term is to compute the reflectance of different views of the light-field. We do not enforce any similarity constraint between reflectance of different views in this term. However, due to joint optimization we get some implicit consistency between views.

Disparity Term: This term enforces neighboring views to have a consistent reflectance. Therefore, we first need to compute correspondences for such view pairs. In [16] they use dense optical flow. However, we use disparity computed from depth wherever it is available, and otherwise use Deqing Sun's implementation of Black and Anandan's dense optical flow method [54], [55]. The disparity term is formulated as:

$$E_d(r) = \sum_m \omega_{mm'}^{occ} (r_m^{(j)} - r_{m'}^{(j+1)})^2 \quad (6)$$

where j is the index of a view in light-field *i.e.*, every two consecutive views contribute to the disparity term. m and m' denote two corresponding pixels. We only constrain those pixels which are not occluded. In case optical flow is used for matching, occlusion map is simply computed using a forward/backward check of the flow field with zero thresholding to enforce consistency. However, in case depth is available, similar to [2] we compute occlusion mask using normalized depth D :

$$\omega_{mm'}^{occ} = \begin{cases} 0.01 & |D_m - D_{m'}| > 0.01 \\ 1 & \text{otherwise} \end{cases} \quad (7)$$

Please note that one might be tempted to use such occlusion masks, to enforce consistency, in the retinex term as well. However, by doing so we will lose information due to missing data in each view. Note that, light-field data is mostly redundant unless there is a large baseline and parallax; therefore, disparity term is most effective in these cases. When reducing computation time is a priority and depth information is not available, user might choose to disable disparity for a dense

diffuse light-field. However, we observed that disparity term improves the quality of the results especially in case of noisy images and flickering illumination. As mentioned in Sec. II-A, while some methods use surface normals information extracted from disparity, we found using a prior on surface normals to be less effective as these estimated normals are noisy and often not effective enough to disambiguate complexities like cast shadows.

Absolute Scale Term: It is widely known in the intrinsic image literature that the absolute scale of intrinsic shading and reflectance layer is ambiguous; and therefore, each method estimates these values up to a constant magnitude. In our particular appearance editing application, extreme shading values (especially black and white shading edges) could result in artifacts when *e.g.*, boosting wrinkles or producing a translucent effect. We solve this by adding a constraint on shading magnitude which prefers moderate values. Such constraint has been used by Xie *et al.* [16] considering only the brightest pixel, or Bell *et al.* [20] constraining every pixel. Constraining the brightest pixel often is not enough, and due to dimensionality of the light-field data considering each pixel is too expensive. Instead, we find constraining 25% of all pixel uniformly sampled from each view to be a good trade-off. Therefore, we formulate this term to penalize extreme shading values on sampled pixels and squeeze them towards a constant $\bar{s} = \log(0.5)$.

$$E_a(r) = \sum_j |s_j - \bar{s}|^2 = \sum_j |i_j - r_j - \bar{s}|^2 \quad (8)$$

L₁ Regularization: We use the method of [56] to further penalize L_1 norm of reflectance. This enforces sparsity in the reflectance image which can replace local and non-local constraints on reflectance introduced in [16]. Final optimization problem can be simplified as the following regularized least squares problem:

$$\min_r \|Tr - d\|_2^2 + \lambda \|r\|_1 \quad (9)$$

T represents the constraint matrix, d is a vector representing the R.H.S of all our constraint equations and r is the unknown reflectance. In all our experiments $\lambda = 0.1$ gives reasonable results. Garces *et al.* [2] use a L_1 filtering as a pre- and post-processing step to improve consistency and reduce noise. Instead we use L_1 regularization as an integral part of our optimization improving the quality of our estimation results.

Many existing methods use a reflectance clustering scheme to further achieve global reflectance sparsity [17], [20], [48], [57]. However, such clustering can be quite sensitive to the choice of number of clusters as well as the effectiveness of the chosen color space and correctness of white-balancing. In general, clustering tends to be more suitable for scenes which contain a limited sparse set of colors. More complex and natural scenes, especially from nature and landscape, often do not follow the global reflectance sparsity assumption. Therefore, in the current work, we take advantage of our edge-based local sparsity scheme.

C. Extension to Dense Light-fields

In case of a dense light-field, the first step is to do a sparse sampling. The intrinsic decomposition of the sparse samples is performed using the method described in Sec. IV-B. The sparse reflectance is then propagated among all the views to get a dense reflectance output. Let us consider that we have sparse reflectance for views at position a and c , given by R_a and R_c respectively.

The reflectance at intermediate position b , say R_b , is obtained by finding a minimizer for the energy functional in Eq. 10.

$$E(R_b) = \int_{\Omega} (1 - \zeta_{mn})(\|\nabla R_b - \nabla I_b\|^2) + w_a(\|R_b - T_{ab}(R_a)\|^2) + w_c(\|R_b - T_{cb}(R_c)\|^2) \quad (10)$$

Please note that ζ_{mn} was used to avoid smoothing of reflectance at reflectance edges (refer Eq. 3 and Eq. 5). In Eq. 10 we use $1 - \zeta_{mn}$ to impose gradient domain constraint only at reflectance edges. In Sec. IV-B we discuss how these edges are obtained. The weights w_a and w_b represent the quality of image mapping. T_{ab} is a warp operator that maps view I_a to I_b and T_{cb} is a warp operator that maps view I_c to I_b . We use the same warp operators to map reflectance R_a and R_c . The energy formulation is based on the idea introduced in [58] and further explored in [59].

However, we have modified this energy for our specific purpose. The first modification is introduced in the form of a weight, $1 - \zeta_{mn}$ for the first energy term. By introducing such a weight, reflectance is enhanced while it is being propagated to dense views. The second modification is making use of two neighboring views, in the angular domain, and their sparse reflectance in the energy formulation. In [59], the authors consider only one previous view in temporal or angular domain. By making use of two nearest neighbors we ensure better consistency among views and also faster convergence for energy minimization. A detailed discussion of the sparse reflectance propagation is presented in the supplementary material.

V. LIGHT-FIELD APPEARANCE EDITING

In this section, we extend band-sifting of the luminance channel as proposed in [12] to independent processing of each intrinsic image layer as derived in Sec. IV. In Sec. V-A we revisit the problem of specular removal. We describe how an approximate *pseudo-specular layer* extraction can be done using band-sifting. In Sec. V-B, we provide a brief description of some of the effects achieved using our sifting-based material and appearance editing framework. We show examples of applying these effects to real-world scenes. In Sec. V-C we compare our editing framework with [12] in terms of quality and robustness for effects like *weathering* and *metallic look*. In Sec. V-B and V-C we use ground-truth intrinsic layers for a fair evaluation of our appearance editing module and demonstrating its strength (see Fig. 3 and 4). All the remaining results in the paper make use of our full framework (including

intrinsic layers). We further introduce new effects such as *pearl* and *translucency* in Sec. V-D.

Based on Eq. 1 we introduce the following notation to describe band-sifting of image I (also refer to Fig. 1):

$$I = S(f_s a_s s_s, \kappa_s) \cdot R(f_r a_r s_r, \kappa_r) + C(f_c a_c s_c, \kappa_c), \quad (11)$$

where $f_l a_l s_l$ represents a component of intrinsic layer $l \in s, r, c$ that undergoes band-sifting. Each component of $f_l a_l s_l$ is characterized by the following signal categories: spatial frequency f_l , signal amplitude a_l and its sign s_l . Similar to [12], we allow only a predefined set of sub-categories: $f_l \in \{H, L, A\}$, $a_l \in \{H, L, A\}$, $s_l \in \{P, N, A\}$, where H and L denote high and low frequency/amplitude range, P and N represent positive and negative values, and A stands for “all”, *i.e.*, the complete category. By controlling the value of multiplication factor κ_l , we can *boost* ($\kappa_l > 1$) or *reduce* ($0 < \kappa_l < 1$) the selected component of the image signal. Additionally, we extend over [12] by supporting negative values of κ_l introducing a new *invert* operation. For some of our material edits we modify the saturation of reflectance layer. The modification is governed by a multiplication factor κ_d , where $\kappa_d > 1$ depicts an increase and $0 < \kappa_d < 1$ implies a decrease in saturation respectively.

Note that in this notation only a single component of each intrinsic layer might undergo band-sifting, and the remaining signal in such layer remains intact. In practice, such a manipulation scope is sufficient to achieve most appearance changes presented in this work. In Sec. VII, we discuss ideas for manipulating multiple components per each intrinsic layer.

An example operation where we *boost* the *high* (H) frequency, *low* (L) amplitude, and *positive* (P) components of *shading* layer by a factor of 4 can be written as: $I = S(HLA, 4) \cdot R + C$. Note that in the above example reflectance and specular layers remain unchanged. We use $L(f_L a_L s_L, \kappa_L)$ notation to indicate when we mean the original band-sifting [12] where an operation is performed on the luminance channel L of the image instead of on its intrinsic layers. We follow the above convention in our examples to denote intrinsic layer modifications and original band-sifting.

A. Highlight/Pseudo-Specular Layer Extraction

In [10], the authors have established relationship between image statistics and the perception of lightness and gloss. Perceptual experiments indicate that by modifying the skewness of sub-bands of luminance histogram, perception of gloss can be altered. In [12], the authors observe that *positive* subband coefficients correspond to bright features like highlights, and the negative values represent features like crevices and holes. We use this observation to first *identify* and then *extract* these bright regions using *sifting* based operations.

The positive component of the sub-bands of the image luminance channel corresponds to its bright regions. We first *sift* the positive components for the complete image (Fig. 2a) to identify the bright regions, using sifting operation: $L(AAP, \kappa_l)$, where $\kappa_l > 1$. We then look for all the pixels which got modified due to this operation to obtain a binary mask (Fig. 2b). In the next step we use *invert* sifting operation

only in the masked region: $L(AAP, \kappa_l)$, where $\kappa_l < 0$. By performing such a step we reduce the highlights present in the image. The difference between the original image and the image with reduced highlights (Fig. 2d) is the *highlight* or the *pseudo-specular* layer (Fig. 2c).

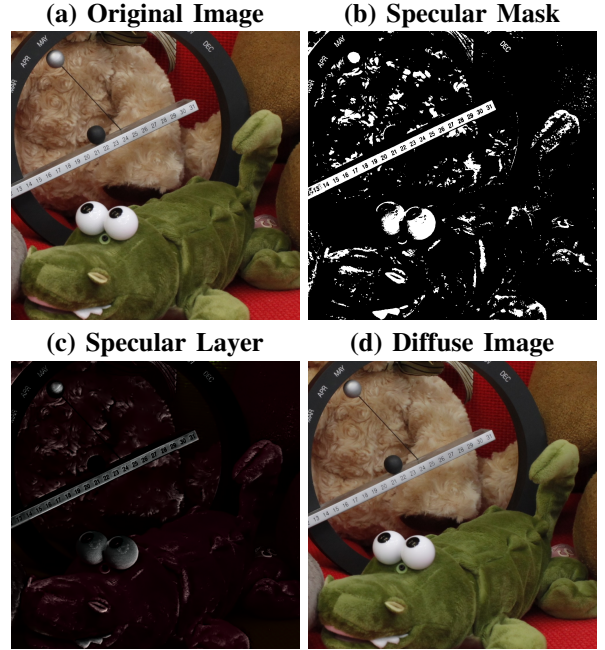


Fig. 2: Specular extraction input and obtained result. Please note that specular layer is rescaled for better visualization. Original image is taken from [60].

B. Types of Appearance Editing

Here, we explore various operations and their effects. We illustrate appearance changes due to editing of shading, reflectance, and specular layers, as well as their combinations.

Intrinsic Shading Layer: of an image contains information on object geometry and illumination. By applying sifting operations on this layer we can enhance or suppress the object shape and geometry details. Here we mostly focus on presenting the outcome of *boost* operation (refer to Sec. V). The *reduce* operation typically leads to the opposite effect. By boosting the high frequency, low amplitude coefficients in the shading layer, we can enhance the fine-level surface details, such as wrinkles and bumps (Fig. 3b).

Intrinsic Reflectance Layer: of an image contains the color and texture information. Similarly to shading, we can apply band-sifting like operations on this layer to enhance color details. By boosting the low-frequency coefficients in the reflectance layer we can make the scene look more vivid (Fig. 3c). Texture colors can be made more pronounced by boosting high frequency coefficients (Fig. 3f).

Intrinsic Specularity Layer: Humans make use of shine and gloss information to classify objects into different categories. By making an object look more shiny, one can make the material appear more plastic or metal like. By boosting all coefficients in the specular layer we can increase the overall

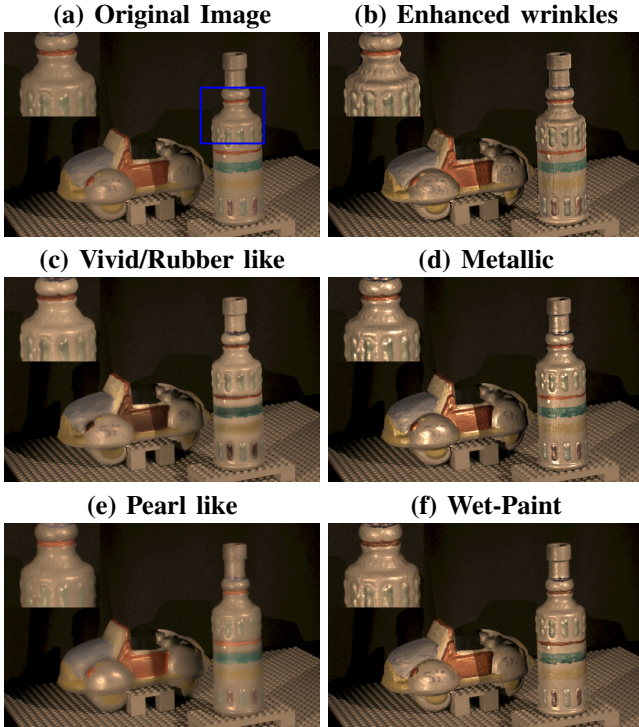


Fig. 3: Different type of material edits using our framework, (b) sift-operation: $S(HLA, 10)$ (c) sift-operation: $R(LLA, 10)$ (d) sift-operation: $C(AAA, 3)$ (e) sift-operation: $S(HHA, -5)$, $R(HAP, -6)$, and $C(LAA, 4)$ (f) sift-operation: $R(HHN, 5.5)$ and $C(HHN, 5.5)$. The top left of each image shows a zoomed in version of the region marked by the blue rectangle in (a). Original image and ground truth intrinsic layers are taken from [61].

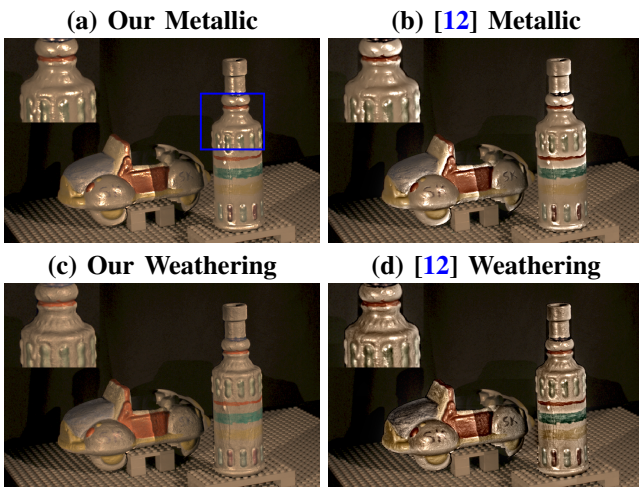


Fig. 4: Comparison of our sifting framework with original luminance-based band-sifting, (a) sift-operation: $S(HHP, 3)$ and $C(HHP, 8)$ (b) sift-operation: $L(HHP, 8)$ (c) sift-operation: $S(HLA, 9)$, $R(AAA, 0.6)$, and $C(AAA, 0.8)$ (d) sift-operation: $L(HLA, 9)$. The top left of each image shows a zoomed in version of the region marked by the blue rectangle in (a).

specularity of the objects, thereby making it look less diffuse and more metallic (Fig. 3d).

Multiple Intrinsic Layers: By sifting the combination of intrinsic layers we can achieve other interesting appearance edits such as wet-paint (Fig. 3f), wet-oily/metallic look (Fig. 4a), and weathering (Fig. 4c). The weathering effect can be further enhanced by desaturating the chroma channels of reflectance layer (see Fig. 14 mid-row)

C. Comparison with Boyadzhiev et al.

Please note that some effects mentioned till now can also be partly achieved by simply sifting the luminance channel of an image. [12]. However, in the latter case only moderate *boost* or *reduce* factors are typically applicable due to possible unnatural look or even explicit artifacts (Fig. 4). The obvious reason for such artifacts is poorer selectivity of edited signal in the luminance channel. As shown in Fig. 4a and c, sifting shading and specularity is more robust than sifting the entire luminance with similar factor. Similarly, more convincing weathered look can be achieved by boosting shading, and reducing reflectance and specularity, which is not possible when a single luminance channel is edited (see Fig. 4d, Fig. 5). Moreover, all our sifting examples are performed on pixel *intensity* channel given by, $In = \sqrt{r^2 + g^2 + b^2}$, where r , g , and b are the color channels, instead of luminance $L = 0.2126r + 0.7152g + 0.0722b$. We observed that by using In instead of luminance L , sifting becomes more robust against color artifacts. In the supplementary material, we provide additional results showing how the range of edits is increased in our case as compared to [12].



Fig. 5: Comparison of making a human face old using our framework with that of the original band-sifting [12]. Please see supplementary for enlarged images. Original image is taken from [12].

D. New Appearance Effects

By making use of multiple intrinsic layers we produce new appearance effects which were not possible by simple 2D image filtering to the best of our knowledge. Please note that attempts have been made to achieve such appearance editing using only images [5] [62]. However, that involved partial to full *inverse-rendering*.

Translucency: is one of the important effects in this category. As proposed by [7], *inverting* the high frequency components of shading layer makes an object appear more translucent. Inspired by their work, we propose techniques which add up to produce a realistic translucent look. By inverting the high frequency coefficients of both shading and

reflectance layers we achieve a certain degree of translucency. We can enhance this effect further by smoothing and desaturating the chroma channels of the reflectance layer. In order to make translucency effect look more realistic we take inspiration from the work of [5], and *inpaint* the background within the object boundaries (Fig. 13 and 6). By refraining from such inpainting and using a more moderate boosting coefficients we can achieve a *pearl-like* look (Fig. 3e). In [63] the authors have discussed how to reconstruct the background, behind occlusions, using *synthetic aperture refocusing*. The *synthetic aperture* technique can also be used for moderately sparse light-fields. In [64] the authors get rid of reflections and occlusions to separate an image into occlusions/reflections and a clear background. For our transparent appearance edit, where the background is typically distorted due to refraction and blurred, a rough approximation of background is enough. We believe that such ideas for background reconstruction can greatly improve the realism of our transparent edits. We leave detailed investigations in this regard as future work.

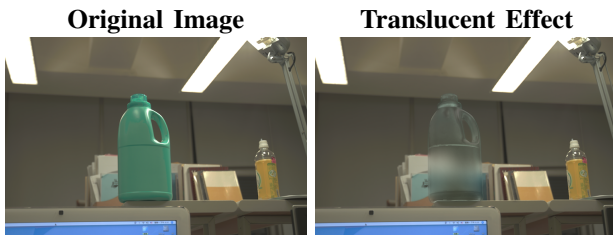


Fig. 6: Translucency effect produced using our intrinsic decomposition and sifting based editing. Original image is taken from [65].

Wetness: appearance edit is inspired from a recent work of Shimano *et al.* [66], where the authors conclude two fundamental characteristics of a wet surface appearance: “darkening and spectral sharpening”. We perform *darkening* on reflectance layer by reducing its intensity, *spectral sharpening* is achieved by increasing the saturation of chroma channels of reflectance layer. We further enhance this effect by amplifying the fine shape details of an object by manipulating the shading layer (Fig. 7).

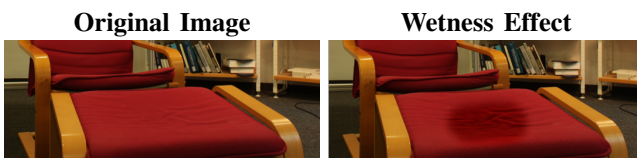


Fig. 7: Wetness effect applied to the chair (from Fig. 9 top-left). Original image is taken from [67].

Depth-guided Selective Filtering: In case of light-field, we can create depth-maps for a given scene using off-the-shelf depth estimation techniques. We can then make use of these depth-maps to selectively target objects at different depths for all the effects mentioned previously *e.g.*, by modulation of editing magnitude (effectively, the coefficient κ in Eq. 11) as a function of depth as demonstrated in Fig. 8. Please note that

the appearance editing is more pronounced for the objects in front.



Fig. 8: Cropped *Couch* scene from the Disney dataset and its wrinkled version that is increasingly blended with the original image as a function of depth.

VI. EVALUATION RESULTS

In this section, we present an evaluation of the two key components of our framework: intrinsic image decomposition and extended band-sifting. To this goal we use five real-world datasets [60], [67]–[70]. Ground-truth intrinsic data is only available for the stereo in [69]. Due to space limit, some of our results are presented in the supplementary materials and videos. We also discuss the computational complexity for all key components of our technique. Finally, we present user interaction scenarios for material appearance editing.

A. Light-field Intrinsic Image Decomposition

Since intrinsic image decomposition is a well-established research problem on its own, in this section we evaluate our decomposition approach as presented in Sec. IV with respect to the state-of-the-art methods for single images [20], RGB-D [15], video [17], stereo-image pairs [16], [51], and light fields [2], [3]. Note that we do not compare against multiview methods like [71], [72] since they use a 360° coverage of the object, which require that either the environment map or the true geometry is given, and each of their scenes contains a single object. We further exclude works like [73] since they require a single known light source, single object, and user-interaction.

Fig. 9 and 10 compare our results with respect to [2], [15]–[17], [20], where typically we produce more consistent and shading-free reflectance for sparsely sampled light-field data. For example in Fig. 10, other methods label reflectance edges as shading specially on the eyes and numbers. Note that our results are not only consistent among views, but also within each view and do not exhibit patch-like artifacts. Recently, Bonneel *et al.* [58] showed promising results on improving consistency across views for the reflectance layer as produced by existing methods. Yet, achieving consistency within an image had remained a challenging task, which we address using our optimization scheme. We further show in the supplementary material and video results that our light-field decomposition is more consistent than simply applying

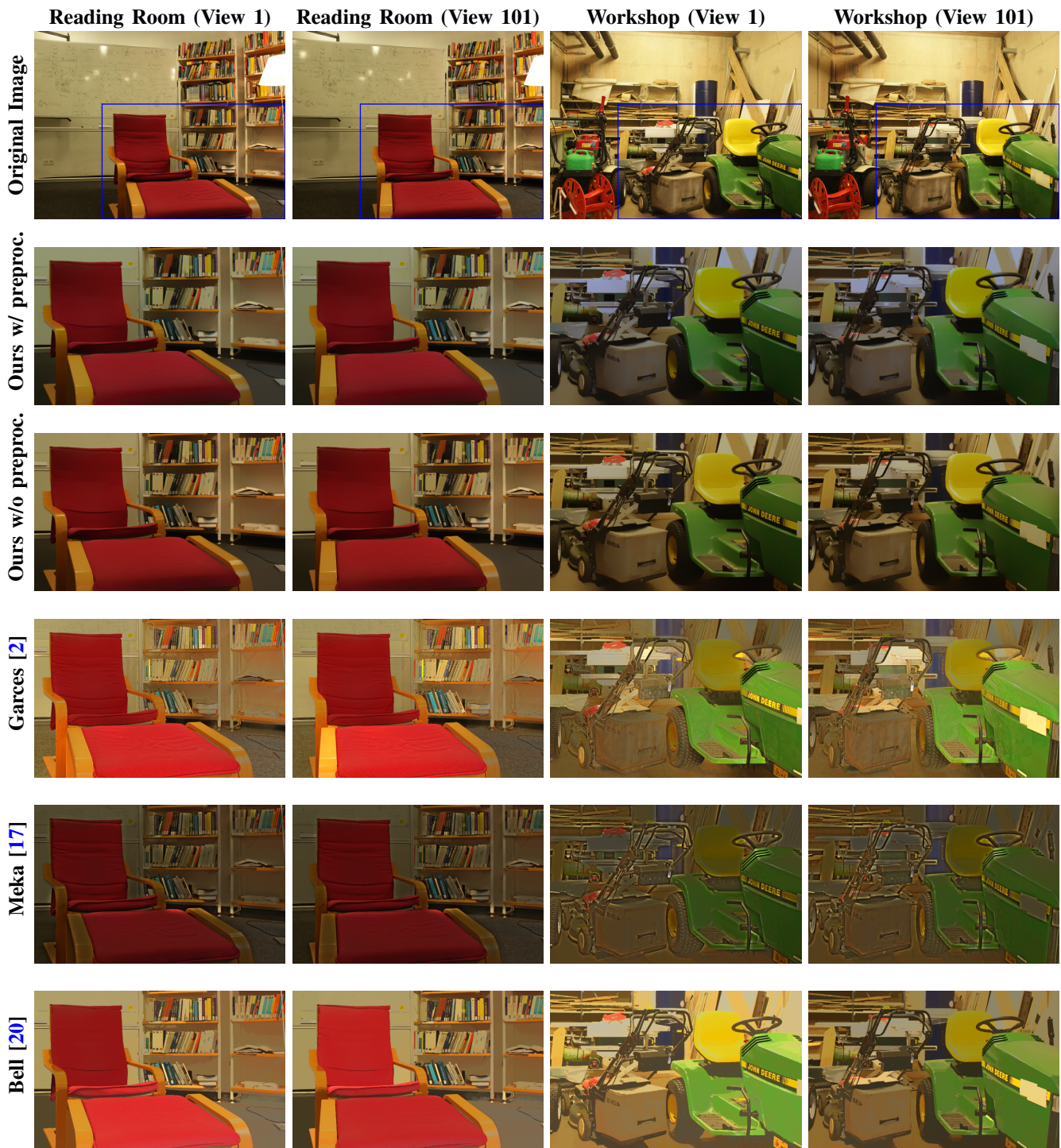


Fig. 9: Comparison of the reflectance layer extraction using our (with/without the pre-processing step consisting of specular removal and white-balancing), Garces *et al.* [48], Meka *et al.* [17], and Bell *et al.* [20] methods. Light-fields for two natural scenes composed of 101 views (1 cm baseline between neighboring views) have been used and only the first and last views are shown. As Garces *et al.* and Meka *et al.* rely on dense data, all 101 views are used. But for our method, we subsample only 11 equally spaced views (10 cm baseline). For Bell *et al.* we perform decomposition separately on each view. While no method is perfect, our reflectance results contain much less shading, specular, and no patch-like artifacts. Our results are consistent not only between views, but also within each view (e.g., the red chair). Please refer to the supplementary materials for the shading results. Original images are taken from [67].

Bonneel *et al.* consistency scheme to the per view intrinsic

decomposition. Fig. 11 shows that unlike [51], in our results the missing correspondence between views due to occlusions does not

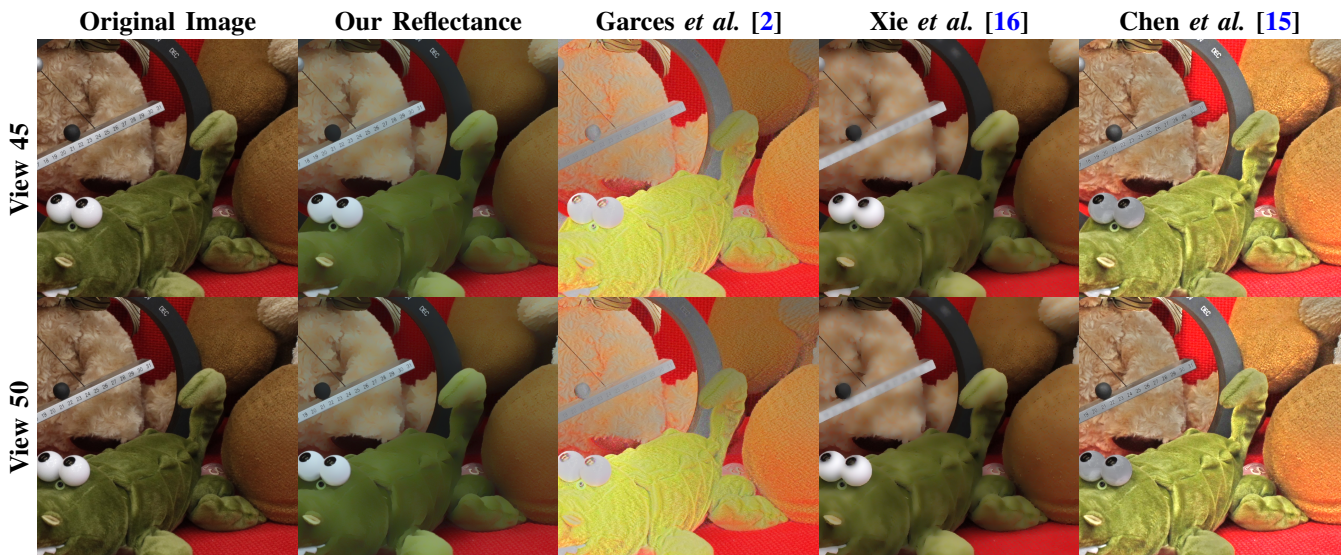


Fig. 10: Comparison of the reflectance layer extraction using our, Garces *et al.*, Xie *et al.*, and Chen *et al.* methods. Each method has a different input: Ours uses 11 subsampled views (10 mm baseline) and given disparity, Garces *et al.* uses 50 dense views (2 mm baseline), Xie *et al.* [16] uses these two views and given optical flow, and Chen *et al.* is performed separately for each view given the existing disparity. Note that contrast is lost (the eyes and text) for Garces *et al.* and Xie *et al.*, there are inconsistency problems for Garces *et al.*, and more shading component remains for Chen *et al.* and Koltun *et al.*

result in artifacts, and we recover consistent reflectance.

Fig. 12 compares our method with [3]. Both [2] and [3] leverage the dense light-field structure and rely on its small baseline, and hence would not handle sparse light fields (10-20 cm baselines). Nevertheless, we compare our performance with these methods by applying our intrinsic image decomposition on a sparse subsample of dense light-field data.

B. Appearance Editing Using Intrinsic Layers

Here we evaluate the performance of our appearance editing framework using intrinsic image layers estimated by our method. Fig. 13 presents our *Translucency* operation on stereo images of [69] (15 cm baseline). We compare the results when using intrinsic layers estimated by our method versus the ground-truth. While our method has misclassified some of the reflectance edges as shading edges, the perceptual quality of the final result has not particularly suffered.

Furthermore, Fig. 14 presents results of applying our full framework using intrinsic layers estimated by our method on three different scenes.

C. Performance

In this section we discuss the performance of our complete pipeline for an example 1D (only horizontal parallax) light-field composed of 101×1 views of spatial resolution 960×720 (Fig. 9 Reading Room).

a) Light-Field Intrinsic Decomposition: As the first pre-processing step we extract the specular layer for each view. It takes 14 – 15 sec. per view for specularity extraction. In the next step we sub-sample the light-field and extract 11 sparse views for intrinsic decomposition. The intrinsic decomposition, including white-balancing, of 11 views takes

70 – 80 min. on average. The sparse results are then propagated across remaining 90 views to get dense results. The propagation step takes 30 – 35 sec. per view on average. Please note that both for intrinsic decomposition and consistent propagation step we assume that we already have the optical flow values for computing correspondence between views. Thus our unoptimized matlab code for intrinsic decomposition and consistent propagation, as a whole, takes approximately 1.5 min. per view. The unoptimized matlab code runs on a Windows 64 bit machine with 32 GB RAM and Intel Xeon CPU (3.50 GHz, 2 processors).

b) Interactive Appearance Editing: Once we have the intrinsic layers of albedo, shading and specularity for each view we can proceed with appearance editing. Our interactive image editing interface, takes as an input a selected view of the light-field and its corresponding intrinsic layers. The GPU implementation of the subband creation takes 1 – 1.5 sec., which is a one time process. The GPU implementation of intrinsic layers sifting, then allows a user to interactively manipulate the image appearance at the rate of 50 – 60 ms per edit. We then apply similar edits on all the views of the light-field, which takes 1 – 1.7 sec per view. The above parallel C++ implementation is based on OpenCV and CUDA (using Nvidia GTX 970 GPU).

D. User Interaction

The interactive interface allows a user to play with different parameters for a single image. Please note that many of the high-level appearance edits (such as “old” look shown in Fig. 5) involve combinations of parameters that remain in certain relation while performing manipulations. To simplify the editing task we provide a single slider to perform such



Fig. 11: Comparison of the reflectance layer extraction using ours (bottom-row) and Maurer *et al.* [51] (top-row) methods on a multi-view stereo dataset. Note that due to occlusion the method by Maurer *et al.* produces artifacts to the right of the fountain. Our results with and without white-balancing show that in some cases (especially outdoors) white balancing the image is not necessary. In our optimization scheme the pre-processing step for white-balancing can be enabled/disabled by setting a flag.

Original image is taken from (fountain-P11 scene) [68].

aggregated parameter changes as demonstrated in the supplemental video. The intuitive interface enables us to quickly find the parameter values for the desired appearance which can later be applied to all the views. Due to our consistent intrinsic layer decomposition, regularity is maintained between views for different appearance modifications.

We could further simplify the user navigation over the parameter choices by developing a collaborative editing system [74] that registers parameter configurations that are often selected by the users. Alternatively, crowd sourcing can be used to learn such meaningful parameter configurations [75]. We could then enable sampling from such parameter distributions to obtain meaningful appearance variations, possibly supported by image-gallery-style user interfaces [76]. We relegate such interaction scenarios as future work.

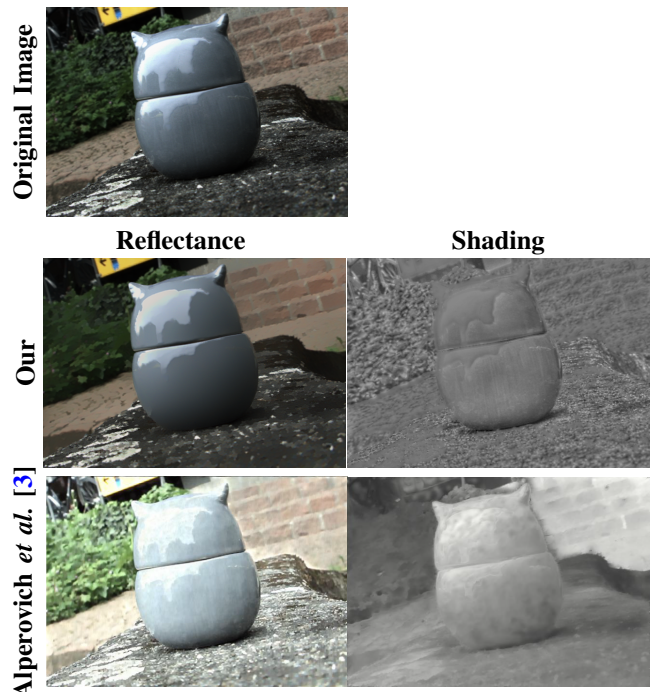


Fig. 12: Comparison of our intrinsic image decomposition with the work of Alperovich and Goldluecke [3]. Original image is taken from [70].

VII. DISCUSSION

Our goal is to handle various possibilities by providing a generic framework which works for different sources of data and different types of objects and materials. All the mentioned appearance editing operations are applicable for single image and video as well. We do not require the user to switch to light-field, however in case of light-fields we assure better quality through per-view consistency. Moreover in case of light-fields, depth computation is straightforward which is necessary for depth-guided appearance editing.

One aspect which we have not considered in our material editing framework is that of applying multiple sifting operations on the same intrinsic layer. For example, $S(HLA, 4)$ and $S(LHN, -3)$. Such operations can be applied in a cascading manner one after another or in a parallel fashion. If done in parallel, we can use different ways to combine these outputs, for example by taking a linear combination.

Limitations: While our translucent appearance effect performs well on the object itself and produces the desired result, cast shadows and reflections of the object on its surrounding also need to be accordingly corrected. Fig. 15 demonstrates a failure case example where the object has taken a translucent appearance that no longer matches its reflection.

Finally, we believe improving specular layer extraction and using a multi-illuminant illumination estimation method to be good avenues for future work.

VIII. CONCLUSION

We present a framework for intrinsic image-based surface appearance editing on wide-baseline light-field data. We



Fig. 13: Translucency results. From left to right: original image; translucency with parameters $S(HAA, -7)$, $R(HAP, -9)$, and $C(LAA, 10)$ using intrinsic layers (reflectance, shading, and specularly) calculated by our own method; and translucency with parameters $S(HAA, -15)$, $R(HAP, -10)$, and $C(LAA, 7)$. While our intrinsic layers are not perfect, we nevertheless achieve a convincing translucent look. Original image and ground-truth intrinsic images are taken from [69].

extract reflectance and shading layers by jointly optimizing on different views to maintain both angular and spatial consistency improving over the state-of-the-art solutions. We present a rich variety of perceptual appearance editing effects by filtering each intrinsic layer separately in terms of frequency, amplitude, and sign. We demonstrate that our intrinsic-image based filtering improves over previous luminance-based solutions in terms of robustness, as well as enables new appearance effects like wetness, translucency, pearlescence. Unlike, full inverse-rendering, we do not require geometry or environment map information. Our modular framework facilitates future extensions. The project web-page with supplementary material and more results is available at <http://light-field-appearance-intrinsic.mpi-inf.mpg.de/>.

IX. ACKNOWLEDGMENT

We would like to thank Osman Ali Mian and Zeeshan Khan Suri for their help during this project. We would like to thank Abhimitra Meka and Elena Garces for kindly providing the necessary comparisons. We thank the reviewers for their insightful comments. The project was supported by the Fraunhofer and Max Planck cooperation program within the German pact for research and innovation (PFI).

REFERENCES

- [1] S. Foessel, F. Zilly, M. Schöberl, P. Schäfer, M. Ziegler, and J. Keinert, "Light-field acquisition and processing system for film productions," in *SMPTE 2013 Annual Technical Conference Exhibition*, Oct 2013, pp. 1–8.
- [2] E. Garces, J. I. Echevarria, W. Zhang, H. Wu, K. Zhou, and D. Gutierrez, "Intrinsic light field images," *Computer Graphics Forum*, vol. 36, no. 3, 2017.
- [3] A. Alperovich and B. Goldluecke, "A variational model for intrinsic light field decomposition," in *Computer Vision – ACCV 2016*, S.-H. Lai, V. Lepetit, K. Nishino, and Y. Sato, Eds. Cham: Springer International Publishing, 2017, pp. 66–82.
- [4] D. C. Bergmann S, Ritschel T, "Interactive appearance editing in rgb-d images," in *Proceedings Vision, Modeling & Visualization*, 2014.
- [5] E. A. Khan, A. Reinhard, R. W. Fleming, and H. H. Bühlhoff, "Image-based material editing," *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 25, no. 3, pp. 654–663, 2006.
- [6] S. Duchêne, C. Riant, G. Chaurasia, J. Lopez-Moreno, P.-Y. Laffont, S. Popov, A. Bousseau, and G. Drettakis, "Multi-view intrinsic images of outdoors scenes with an application to relighting," *ACM Transactions on Graphics*, vol. 34, no. 5, pp. 164:1–164:16, 2015.
- [7] R. W. Fleming and H. H. Bühlhoff, "Low-level image cues in the perception of translucent materials," *ACM Transactions on Applied Perception*, vol. 2, no. 3, pp. 346–382, 2005.
- [8] R. Fleming, "Human perception: Visual heuristics in the perception of glossiness," *Current Biology*, vol. 22, no. 20, pp. 865–866, 2012.
- [9] I. Motoyoshi, S. Nishida, L. Sharan, and E. Adelson, "Image statistics and the perception of surface qualities," vol. 447, pp. 206–9, 06 2007.
- [10] L. Sharan, Y. Li, I. Motoyoshi, S. Nishida, and E. H. Adelson, "Image statistics for surface reflectance perception," *Journal of the Optical Society of America A*, vol. 25, no. 4, pp. 846–865, 2008.
- [11] M. Giesel and Q. Zaidi, "Frequency-based heuristics for material perception," *Journal of Vision*, vol. 13, no. 14, pp. 7–7, 2013.
- [12] I. Boyadzhiev, K. Bala, S. Paris, and E. Adelson, "Band-sifting decomposition for image-based material editing," *ACM Transactions on Graphics*, vol. 34, no. 5, 2015.
- [13] H. Barrow and J. Tenenbaum, "Recovering intrinsic scene characteristics from images," *Computer Vision Systems*, A. Hanson and E. Riseman (eds.), Academic Press, New York, pp. 3–26, 1978.
- [14] N. Bonneel, B. Kovacs, S. Paris, and K. Bala, "Intrinsic decompositions for image editing," in *Computer Graphics Forum*, vol. 36, no. 2, 2017, pp. 593–609.
- [15] Q. Chen and V. Koltun, "A simple model for intrinsic image decomposition with depth cues," in *Proceedings IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 241–248.
- [16] D. Xie, S. Liu, K. Lin, S. Zhu, and B. Zeng, "Intrinsic decomposition for stereoscopic images," in *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 1744–1748.
- [17] A. Meka, M. Zollhöfer, C. Richardt, and C. Theobalt, "Live intrinsic video," *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 35, no. 4, 2016.
- [18] P.-Y. Laffont, A. Bousseau, and G. Drettakis, "Rich intrinsic image decomposition of outdoor scenes from multiple views," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 2, pp. 210–224, 2013.
- [19] A. Artusi, F. Banterle, and D. Chetverikov, "A survey of specularly removal methods," in *Computer Graphics Forum*, vol. 30, no. 8, 2011, pp. 2208–2230.
- [20] S. Bell, K. Bala, and N. Snavely, "Intrinsic images in the wild," *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 33, no. 4, pp. 159:1–159:12, 2014.
- [21] M. W. Tao, J.-C. Su, T.-C. Wang, J. Malik, and R. Ramamoorthi, "Depth estimation and specular removal for glossy surfaces using point and line consistency with light-field cameras," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 38, no. 6, pp. 1155–1169, 2016.
- [22] A. Sulc, A. Alperovich, N. Marniok, and B. Goldluecke, "Reflection separation in light fields based on sparse coding and specular flow," in *Proceedings of the Conference on Vision, Modeling & Visualization*, 2016, pp. 137–144.
- [23] Y. Gryaditskaya, B. Masia, P. Didyk, K. Myszkowski, and H.-P. Seidel, "Gloss editing in light fields," in *Conference on Vision, Modeling & Visualization (VMV)*, 2016, pp. 127–135.
- [24] A. Chadwick and R. Kentridge, "The perception of gloss: a review," *Vision Research*, vol. 109, pp. 221–235, 2015.
- [25] Z. Farbman, R. Fattal, D. Lischinski, and R. Szeliski, "Edge-preserving decompositions for multi-scale tone and detail manipulation," *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 27, no. 3, pp. 67:1–67:10, 2008.
- [26] R. Fattal, M. Agrawala, and S. Rusinkiewicz, "Multiscale shape and detail enhancement from multi-light image collections," *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 26, no. 3, 2007.
- [27] S. Paris, S. W. Hasinoff, and J. Kautz, "Local laplacian filters: Edge-aware image processing with a laplacian pyramid," *ACM Transactions*

- on *Graphics (Proceedings SIGGRAPH)*, vol. 30, no. 4, pp. 68:1–68:12, 2011.
- [28] E. Gastal and M. Oliveira, “Domain transform for edge-aware image and video processing,” *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 30, no. 4, pp. 69:1–69:12, 2011.
- [29] S. Bae, S. Paris, and F. Durand, “Two-scale tone management for photographic look,” *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 25, no. 3, pp. 637–645, 2006.
- [30] Q. Chen, J. Xu, and V. Koltun, “Fast image processing with fully-convolutional networks,” *arXiv:1709.00643*, vol. cs.CV, Sep. 2017.
- [31] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu, “Light field image processing: An overview,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 7, pp. 926–954, Oct 2017.
- [32] C. Birklbauer and O. Bimber, “Light-field retargeting,” in *Computer Graphics Forum*, vol. 31, no. 2, 2012, pp. 295–303.
- [33] B. Chen, E. Ofek, H.-Y. Shum, and M. Levoy, “Interactive deformation of light fields,” in *Proc. ACM i3D*, ser. i3D ’05, 2005, pp. 139–146.
- [34] O. Frigo and C. Guillemot, “Epipolar plane diffusion: An efficient approach for light field editing,” in *Proceedings of British Machine Vision Conference (BMVC)*, Sep. 2017.
- [35] D. R. Horn and B. Chen, “Lightshop: Interactive light field manipulation and rendering,” in *Proceedings of the 2007 ACM Symposium on Interactive 3D Graphics and Games*, ser. i3D ’07, 2007, pp. 121–128.
- [36] Z. Zhang, L. Wang, B. Guo, and H.-Y. Shum, “Feature-based light field morphing,” *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 21, no. 3, pp. 457–464, 2002.
- [37] S. Seitz and K. Kutulakos, “Plenoptic image editing,” *International Journal of Computer Vision*, vol. 48, no. 2, pp. 115–129, 2002.
- [38] A. Jarabo, B. Masia, A. Bousseau, F. Pellacini, and D. Gutierrez, “How do people edit light fields?” *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 33, no. 4, 2014.
- [39] William, K. W. Shon, and I. K. Park, “Spatio-angular consistent editing framework for 4d light field images,” *Multimedia Tools and Applications*, vol. 75, no. 23, pp. 16615–16631, Dec 2016.
- [40] A. Jarabo, B. Masia, and D. Gutierrez, “Efficient propagation of light field edits,” in *Proc. of the V Ibero-American Symposium in Computer Graphics*, ser. SIACG 2011, 2011, pp. 75–80.
- [41] H. Sun, P. Li, and B. Sheng, “Image-based material restyling with fast non-local means filtering,” *International Conference on Image and Graphics*, pp. 841–846, 2009.
- [42] R. Vergne, P. Barla, R. W. Fleming, and X. Granier, “Surface flows for image-based shading design,” *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 31, no. 4, pp. 94:1–94:9, 2012.
- [43] S. Xuey, J. Wang, X. Tong, Q. Dai, and B. Guo, “Image-based material weathering,” *Computer Graphics Forum*, 2008.
- [44] S.-K. Yeung, C.-K. Tang, M. S. Brown, and S. B. Kang, “Matting and compositing of transparent and refractive objects,” *ACM Transactions on Graphics*, vol. 30, no. 1, pp. 2:1–2:13, 2011.
- [45] F. Di Renzo, C. Calabrese, and F. Pellacini, “Appim: Linear spaces for image-based appearance editing,” *ACM Transactions on Graphics (Proceedings SIGGRAPH Asia)*, vol. 33, no. 6, pp. 194:1–194:9, 2014.
- [46] R. O. Dror, E. H. Adelson, and A. S. Willsky, “Recognition of surface reflectance properties from a single image under unknown real-world illumination,” in *Proc. of IEEE Workshop on Identifying Objects Across Variations in Lighting*, 2001, pp. 1–8.
- [47] Y. Ostrovsky, P. Cavanagh, and P. Sinha, “Perceiving illumination inconsistencies in scenes,” *Perception*, vol. 34, no. 11, 2005.
- [48] G. Ye, E. Garces, Y. Liu, Q. Dai, and D. Gutierrez, “Intrinsic video and applications,” *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 33, no. 4, pp. 80:1–80:11, 2014.
- [49] R. Grosse, M. Johnson, E. Adelson, and W. Freeman, “Ground truth dataset and baseline evaluations for intrinsic image algorithms,” in *IEEE ICCV*, 2009, pp. 2335–2342.
- [50] J. Van De Weijer, T. Gevers, and A. Gijsenij, “Edge-based color constancy,” *IEEE Transactions on Image Processing*, vol. 16, no. 9, pp. 2207–2214, 2007.
- [51] D. Maurer, Y.-C. Ju, M. Breuß, and A. Bruhn, “Combining shape from shading and stereo: a variational approach for the joint estimation of depth, illumination and albedo,” in *Proceedings of British Machine Vision Conference (BMVC)*, 2016.
- [52] Q. Zhao, P. Tan, Q. Dai, L. Shen, E. Wu, and S. Lin, “A closed-form solution to retinex with nonlocal texture constraints,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 34, no. 7, pp. 1437–1444, 2012.
- [53] J. Van De Weijer, T. Gevers, and A. W. Smeulders, “Robust photometric invariant features from the color tensor,” *IEEE Transactions on Image Processing*, vol. 15, no. 1, pp. 118–127, 2006.
- [54] D. Sun, S. Roth, J. P. Lewis, and M. J. Black, *Learning Optical Flow*. Springer Berlin Heidelberg, 2008.
- [55] M. J. Black and P. Anandan, “The robust estimation of multiple motions,” *Computer Vision and Image Understanding*, vol. 63, no. 1, pp. 75–104, 1996.
- [56] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, “An interior-point method for large-scale l1 regularized least squares,” *IEEE Journal of Selected Topics in Signal Proc.*, vol. 1, no. 4, pp. 606–617, 2007.
- [57] P. V. Gehler, C. Rother, M. Kiefel, L. Zhang, and B. Schölkopf, “Recovering intrinsic images with a global sparsity prior on reflectance,” in *Proceedings of the 24th International Conference on Neural Information Processing Systems*, ser. NIPS’11, 2011, pp. 765–773.
- [58] N. Bonneel, J. Tompkin, K. Sunkavalli, D. Sun, S. Paris, and H. Pfister, “Blind video temporal consistency,” *ACM Transactions on Graphics (Proceedings SIGGRAPH Asia)*, vol. 34, no. 6, pp. 196:1–196:9, 2015.
- [59] N. Bonneel, J. Tompkin, D. Sun, O. Wang, K. Sunkavalli, S. Paris, and H. Pfister, “Consistent video filtering for camera arrays,” *Computer Graphics Forum*, vol. 36, no. 2, pp. 397–407, 2017.
- [60] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. Gross, “Scene reconstruction from high spatio-angular resolution light fields,” *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 32, no. 4, pp. 73:1–73:12, 2013.
- [61] S. Beigpour, A. Kolb, and S. Kunz, “A comprehensive multi-illuminant dataset for benchmarking of intrinsic image algorithms,” in *Proceedings IEEE International Conference on Computer Vision (ICCV)*, December 2015, pp. 172–180.
- [62] D. Gutierrez, F. J. Seron, J. Lopez-Moreno, M. P. Sanchez, J. Fandos, and E. Reinhard, “Depicting procedural caustics in single images,” *ACM Transactions on Graphics (Proceedings SIGGRAPH Asia)*, vol. 27, no. 5, pp. 120:1–120:9, Dec. 2008.
- [63] V. Vaish, M. Levoy, R. Szeliski, C. L. Zitnick, and S. B. Kang, “Reconstructing occluded surfaces using synthetic apertures: Stereo, focus and robust measures,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, ser. CVPR ’06, 2006, pp. 2331–2338.
- [64] T. Xue, M. Rubinstein, C. Liu, and W. T. Freeman, “A computational approach for obstruction-free photography,” *ACM Transactions on Graphics (Proceedings SIGGRAPH)*, vol. 34, no. 4, 2015.
- [65] C.-K. Liang and C. Chung. (2006) Image-based material editing. [Online]. Available: http://chiakailiang.org/project_ibme/
- [66] M. Shimano, H. Okawa, Y. Asano, R. Bise, K. Nishino, and I. Sato, “Wetness and color from a single multispectral image,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 321–329.
- [67] V. K. Adhikarla, M. Vinkler, D. Sumin, R. K. Mantiuk, K. Myszkowski, H. Seidel, and P. Diddy, “Towards a quality metric for dense light fields,” in *CVPR*, 2017, pp. 3720–3729.
- [68] C. Strecha, W. Von Hansen, L. Van Gool, P. Fua, and U. Thoennessen, “On benchmarking camera calibration and multi-view stereo for high resolution imagery,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Ieee, 2008, pp. 1–8.
- [69] S. K. A. K. Shida Beigpour, Mai Lan Ha and V. Blanz, “Multi-view multi-illuminant intrinsic dataset,” in *Proceedings of the British Machine Vision Conference (BMVC)*, 2016, pp. 10.1–10.13.
- [70] S. Wanner, S. Meister, and B. Goldlücke, “Datasets and benchmarks for densely sampled 4d light fields,” in *Conference on Vision, Modeling & Visualization*, 2013, pp. 225–226.
- [71] G. Oxholm and K. Nishino, “Shape and reflectance estimation in the wild,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 38, no. 2, pp. 376–389, 2016.
- [72] S. Lombardi and K. Nishino, “Reflectance and illumination recovery in the wild,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 38, no. 1, pp. 129–141, 2016.
- [73] T.-C. Wang, M. Chandraker, A. A. Efros, and R. Ramamoorthi, “Svbrdf-invariant shape and reflectance estimation from light-field cameras,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 5451–5459.
- [74] J. O. Talton, D. Gibson, L. Yang, P. Hanrahan, and V. Koltun, “Exploratory modeling with collaborative design spaces,” *ACM Transactions on Graphics (Proceedings SIGGRAPH Asia)*, vol. 28, no. 5, pp. 167:1–167:10, 2009.
- [75] Y. Koyama, D. Sakamoto, and T. Igarashi, “Crowd-powered parameter analysis for visual design exploration,” in *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST ’14, 2014, pp. 65–74.

- [76] J. Marks, B. Andalman, P. A. Beardsley, W. Freeman, S. Gibson, J. Hodgins, T. Kang, B. Mirtich, H. Pfister, W. Ruml, K. Ryall, J. Seims, and S. Shieber, "Design galleries: A general approach to setting parameters for computer graphics and animation," in *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '97, 1997, pp. 389–400.

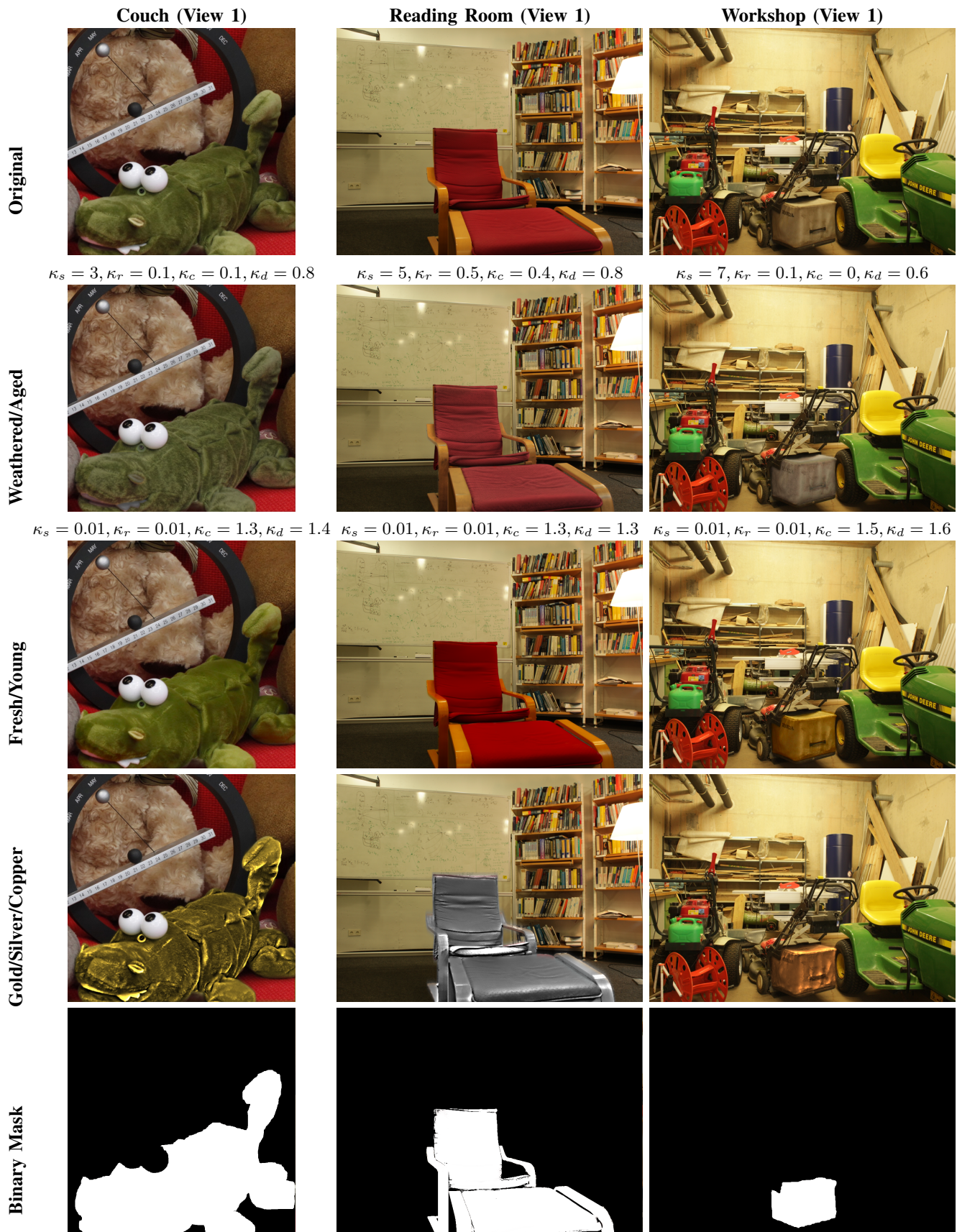


Fig. 14: Our full-framework results for the appearance edit of *weathering* and its reverse effect imparting a *fresh* look in the masked region. In case of weathering we apply sift operation like: $S(HLP, \kappa_s)$ where $\kappa_s > 1$ to enhance wrinkles like fine shape details, $R(AAA, \kappa_r), C(AAA, \kappa_c)$ where $0 < \kappa_r, \kappa_c < 1$ to reduce reflectance and specular brightness and desaturation of chroma channels of reflectance by a factor say κ_d where $0 < \kappa_d < 1$. In case of fresh look we apply sift operation like: $S(HLA, \kappa_s), R(HLA, \kappa_r)$, where $0 < \kappa_r, \kappa_s < 1$ to smooth shading and reflectance, $C(AAA, \kappa_c)$ where $\kappa_c > 1$ for increasing the shine of the object and finally the saturation of chroma channels of reflectance is increased by a factor of κ_d , where $\kappa_d > 1$. Please look at the figure header for exact values used in this case.



Fig. 15: Translucency effect produced using our full framework. Please note that the shadow of the object does not look proper along with its translucent counterpart. Original image is taken from [65].