Do-It-Yourself Lighting Design for Product Videography

Ivaylo Boyadzhiev
Cornell University
iboy@cs.cornell.edu

Jiawen Chen
Google
jiawen@google.com

Sylvain Paris
Adobe
sparis@adobe.com

Kavita Bala
Cornell University
kb@cs.cornell.edu

Abstract

The growth of online marketplaces for selling goods has increased the need for product photography by novice users and consumers. Additionally, the increased use of online media and large-screen billboards promotes the adoption of videos for advertising, going beyond just using still imagery.

Lighting is a key distinction between professional and casual product videography. Professionals use specialized hardware setups, and bring expert skills to create good lighting that shows off the product's shape and material, while also producing aesthetically pleasing results.

In this paper, we introduce a new do-it-yourself (DIY) approach to lighting design that lets novice users create studio quality product videography. We identify design principles to light products through emphasizing highlights, rim lighting, and contours. We devise a set of computational metrics to achieve these design goals. Our workflow is: the user acquires a video of the product by mounting a video camera on a tripod and using a tablet to light objects by waving the tablet around the object. We automatically analyze and split this acquired video into snippets *that match our design principles. Finally, we present an interface that lets users easily select snippets with specific characteristics and then assembles them to produce a final pleasing video of the product. Alternatively, they can rely on our template mechanism to automatically assemble a video.*

1. Introduction

Popular online marketplaces like eBay, craigslist, and Etsy allow everyone to directly sell their own goods. As a consequence, product photography, a domain that used to be reserved to professionals, is now also needed by novice users. Further, the online nature of these platforms favors the sharing of videos and animated gifs. But producing a professional quality product video is challenging and requires a studio and specialized equipment for lighting. Professionals carefully arrange lights and reflectors to emphasize the product's shape and material while achieving visually compelling video. Setting up such illumination requires time, money, and a great deal of expertise. For these reasons, novices are unable to produce nearly as good a result, and their videos typically look unflattering, and rarely do justice to the product.

Our work aims to let casual users create quality product videos. We propose a simple do-it-yourself (DIY) setup with a video camera on a tripod recording the product while the user waves a tablet around it for a couple of minutes. Although the raw footage is too long and unappealing as is, we show that we are able to identify good *snippets* and assemble them into a short pleasing video that showcases the product. First, we study professional product videos and the literature to formulate the design principles that guide our approach. For instance, video artists always use long light sweeps to create highlights that move consistently over the product. They also rely on a few typical standard illumination configurations such as rim lighting and highlights which follow the main edges of the object, e.g., [25].

We build analysis tools to achieve these design principles. We define scoring functions to rank video segments according to characteristics such as the presence of rim lighting or their emphasis on edges. Robustness is a paramount requirement for these functions because the objects in which we are interested are often highly non-Lambertian (e.g., a transparent and refractive perfume bottle or a bejeweled watch) and many standard analysis algorithms fail on them. We show that our numerical schemes perform well on a wide variety of objects and materials despite these challenges.

We present a graphical interface based on faceted search that lets users select and assemble snippets and produce a video with sophisticated lighting effects. It also provides tools to create a "base image" corresponding to the static illumination on top of which the highlights move. Capturing the input footage takes about 10 minutes, and editing the final result with our interface, another 15 minutes. We show that novices can go through this process with minimal training and produce quality product videos that they would not be able to produce otherwise. Alternatively, we propose a few templates that enable users to automatically assemble a video according to a predefined sequence of shots.

Contributions. To assist novice users with the production of product videos, we make the following contributions:
– A set of design principles for product videography that we support with empirical observations of professional videos and existing literature on lighting design.
– A simple acquisition procedure to generate short video snippets covering a wide variety of light configurations.
– Robust analysis tools to rank these snippets according to the criteria that we expressed in our design principles.
– A faceted-search user interface to browse the snippets and assemble a few of them to produce a quality video.

1.1. Related Work

A few techniques exist to relight video content. For instance, Shih et al. [22] adjust the low-frequency illumination of video portraits as part of their style transfer technique. In comparison, we seek a more fine-grained and more generic technique to control the lighting in product videos. Wenger et al. [26] offer such control but requires a dedicated light stage, which limits its use to professional productions.

More work has been done on the static case. Akers et al. [3], Agarwala et al. [1], and Mohan et al. [18] capture several photos of an object under different illuminations and describe a user interface to combine these images to achieve a desired lighting. Compared to our approach, these methods assume that users are able to guide the algorithm with scribbles whereas we provide high-level design principles and algorithmic tools to help users follow them. Boyadzhiev et al. [6] provide such high-level assistance but their technique does not handle specular objects well. Srikanth et al. [23] help photographers with the positioning of a light source to create and control rim lighting using a drone. While rim lighting is relevant to our work, we are also interested in several other effects and aim for a simpler setup not requiring a drone. Fattal et al. [8] also use a multi-light image collection and describe an image fusion algorithm to create a single image that reveals fine details. This technique is effective, but user control is limited to a couple of predefined settings whereas we seek to give more creative control to users. Winnemöller et al. [27] let users directly specify an environment map that they are able to approximate by estimating the position of each light in the input collection. While specifying an envmap gives full control over the illumination, it requires expert users that are able to specify this information. Bousseau et al. [5] assist users in this task with an algorithm that optimizes an envmap for a given effect and material, e.g., to maximize back lighting on a wax candle. However, their approach requires full 3D knowledge of the scene and its materials, which is impractical in our context. Lopez-Moreno et al. [16; 15] and Karsch et al. [12] address the problem of inferring lighting directions and rough geometry from a single photo. Those systems allow the applications of relighting and inclusion of new objects in existing static scenes, but the user still has to specify the target lighting. In comparison we propose high-level principles that guide the target lighting for product photography based on capturing video of a static object under dynamic lighting, thus avoiding the potentially brittle stage of estimating 3D geometry for the purpose of relighting. Further, recall that all techniques in this paragraph focus on static images, whereas we are interested in videos.

Finally, our approach is also related to techniques that select video snippets either to generate a summary, e.g., [2], or an infinitely looping video, e.g., [21; 14]. However, these techniques are concerned with the duration of the video and are not about lighting design.

1.2. Overview

Our new workflow comprises three parts: acquisition, analysis, and compositing.

Acquisition. The user records a short video (3-4 minutes) by mounting a camera on a tripod and waving an area light source around the object of interest. We instruct users to wave the light source along smooth arcs approximately centered on the object and to vary the trajectories, e.g., front to back, side to side going above or behind the objects, and other intermediate options. The goal of the acquisition step is to capture sufficiently diverse trajectories so that our snippet selection algorithm extracts enough useful snippets. We recommend that the camera is level with the object and that the object is centered in the frame. While other configurations are possible, this setup is simple and clear to novices. The whole acquisition process takes at about 10 minutes to set up and capture 3-4 minute long videos.

Analysis. The video is analyzed for features such as the speed and direction of the motion of lighting. These features are then used by various metrics to split the video into a set of snippets. Our metrics aim to capture design principles such as highlighting contours, rim lighting, and accentuating meso-structure.

GUI and Compositing. Finally, the user explores the collection of extracted snippets, and composites and sequences them in our GUI to produce the final video.

2. Design Principles

Lighting designers balance many goals when setting up lights. They illuminate the object to show off its material, emphasize its shape, reveal subtle details, while also producing a visually pleasing video. Artists have acquired an intimate understanding of these objectives and their interactions, but we could not find any formal comprehensive description of this craft. Instead, we analyzed professionally produced clips such as [25] [24] and reinterpreted photographic guidelines in the context of product videos to formulate the following principles.

Lighting Properties. The majority of the clips that we analyzed use white area light sources. The lights are either fixed to create a base illumination, or move slowly along long smooth trajectories to generate highlights that move predictably. Further, it has been demonstrated [9] that area lights are better than point lights for material perception; a swept light integrated over time creates an area light effect.

Video Structure. Product videos are typically made up of 4 to 8 shots, each lasting typically between 2 and 10 seconds. There is little to no camera motion during each shot and the light does not change speed or direction. The first shots are framed to depict the entire object or its main part, e.g., the face of a watch. A recurring choice is to use rim lighting on these first shots to show the object silhouette without revealing its appearance. Then, subsequent shots are progressively framed tighter on small details, and the last shot is often a well-lit view of the product in its entirety. In all the videos, to avoid distracting the viewer, the object is shown on top of a simple uncluttered background, often black or white.

Shape and Material. Video artists often use the same strategies as photographers to emphasize the shape and material of a product. An exception is the use of a slowly moving light at a grazing angle to generate glittering on surfaces with specular micro-geometry. Besides glittering, several other effects can be interpreted as an adaptation of well-documented guidelines used for static photography.

Placing lights around the object to maximize the contrast around edges helps reveal the shape of the object [6]. Placing them behind produces rim lighting that shows off the silhouette and separates the object from the background [23]. Setting the light behind a glass object with black elements on the side creates a "bright field" that reveals the shape of the object that would otherwise be transparent [5].

For translucent objects, back and side lighting emphasize the translucency of their material by maximizing scattering [28]. Grazing illumination increases the visibility of the mesostructure of rough surfaces by generating fine-scale shadows [20]. For specular objects, an illumination that minimizes highlights while maximizing the diffuse reflection reveals their intrinsic color [6], while lighting them so that highlights align with the main curved regions helps understand their shape and emphasizes the material's shininess [13].

3. Analysis

The input of our approach is a video sequence taken from a fixed viewpoint while the user slowly moves a light source around the object following long arcs. As is, the raw footage is not usable but it contains many useful subsequences. Our goal is to find them and use them as building blocks for the final video. Our approach also eases the burden of acquisition on users since it is robust to bad segments in the recording.

In this section, we first split the captured footage into shorter segments that we call *snippets*. Then, we analyze these snippets, yielding a set of scores that allow us to rank them according to the criteria derived from our design principles (Section 2). We assume that a foreground mask is available. In practice, it was created by hand, which was easy since the static object stands in front of an uncluttered background.

3.1. Splitting the Input Footage into Snippets

Following our design principles, we seek to isolate long portions where the light follows a smooth trajectory. Since we do not have access to the 3D position of the light during the recording, we rely on image cues to infer this information. Intuitively, smooth light motion results in smooth variations of image properties such as pixel color and optical flow, which one can analyze to infer information about the light. However, image variations can also be triggered by other phenomena such as occlusions and geometric discontinuities and makes relying on a single cue brittle. Instead, we rely on several criteria and introduce a cut point in the video when they collectively indicate a discontinuity. Our rationale is: from our experiments, sharp variations in the light trajectory affect all the cues *simultaneously*, whereas other causes perturb only one or two.

Direction smoothness score. We observe the highlights and estimate how fast the direction of their motion is changing at each frame. We use the Lucas-Kanade method [17] to compute the flow at each pixel between each pair of adjacent frames. While Lucas-Kanade is not the best performing on standard optical flow benchmark, unlike other approaches, it assumes very little about the scene, which is critical to track highlights that typically violate the assumptions made by other techniques (e.g., our objects are not Lambertian). We experimented with other state of the art optical-flow algorithms and found that Lucas-Kanade gives more stable and predictable results on our challenging datasets. We further use the per-pixel confidence values, produced by the algorithm, to concentrate our analysis on pixels where the algorithm behaves well.

First, we estimate the dominant motion direction of the highlights between frames i and $i+1$ by building a histogram of flow vector directions, but only for pixels with a confidence in the top 5%. Each sample is weighted by the magnitude of the optical flow and the intensity of its corresponding pixel, $\bar{I} = 0.299R + 0.587G + 0.114B$. This weight gives more importance to highlights, while reducing that of small flow vectors more likely to be due to noise. We define the dominant direction as the label of the fullest his-

togram bin H_1 and estimate a confidence factor $1 - H_2/H_1$ that favors cases where the selected bin is unambiguously larger than the second fullest bin H_2.

For frame i, we consider the dominant directions of the $N = 12$ previous frames. We build a new histogram with them, this time using their confidence factor as weight and again extract the dominant direction which we call D_ℓ. We do the same with the N following frames to get D_r and compute the angle difference $D_{\ell\mathrm{r}} = [(D_\ell - D_\mathrm{r} + \pi) \mod 2\pi] - \pi$. The direction smoothness score is computed as: $S_\mathrm{d}(i) = \exp(-D_{\ell\mathrm{r}}^2/(\pi/6))$. We found the scale factor $\pi/6$ to work well in our experiments although its exact value had a limited impact on the results. The same is true for the other constants used in the rest of the section.

Highlight speed smoothness score. We now estimate how smoothly the speed of the highlights varies. First, we compute their speed between frames i and $i+1$ as the average of the magnitudes of the flow vectors weighted by the intensity of their corresponding pixel. When computing this average, we discard the amplitudes smaller than 1 pixel because they are likely to be dominated by noise and such small motion is not perceivable. We then compute the median of the N previous and N following frames to get V_ℓ and V_r respectively, and compute the smoothness score $S_\mathrm{a}(i) = \exp((1 - \frac{\min(V_\ell, V_\mathrm{r})}{\max(V_\ell, V_\mathrm{r}) + \epsilon})^2/0.5)$ with $\epsilon = 10^{-7}$.

Light speed smoothness score. For the last cue, we seek to estimate how fast the light was moving when the video was recorded. Our approach is inspired by the work of Winnemöller et al. [27] who showed how image color differences relates to 3D light distances. Inspired by this result, we estimate the speed of the light source between frames i and $i+1$ as the sum of the absolute values of the temporal derivatives at each pixel. Then, similar to the previous case, we compute the medians T_ℓ and T_r and the smoothness score is: $S_\mathrm{s}(i) = \exp((1 - \frac{\min(T_\ell, T_\mathrm{r})}{\max(T_\ell, T_\mathrm{r}) + \epsilon})^2/0.3)$.

Overall snippet smoothness score. Finally, we add all three scores to get $S_\text{cut}(i) = S_\mathrm{d}(i) + S_\mathrm{a}(i) + S_\mathrm{s}(i)$. Low smoothness values are likely to correspond to irregular light motion, e.g., between two sweeps, and local minima are good candidates to cut if needed. Our cutting procedure processes the video recursively, starting with the entire recorded video. Given a sequence of frames $[a, b]$, we first check that it is longer than the threshold L_min, controlling the shortest sequence that we can cut. If it is longer, we compute $\sum_{i\in[a;b]} S_\text{cut}(i) / \min_{i\in[a;b]} S_\text{cut}(i)$ and compare it to the threshold L_max that defines the maximum length of a snippet. If it is above, we cut at the local minimum $\arg\min_{i\in[a;b]} S_\text{cut}(i)$. The advantage of this criterion is that it always cuts sequences longer than L_max and for shorter sequences, the shorter they are, the less likely they are to be cut because the sum in the numerator contains fewer terms. That is, our criterion balances the requirements of not having overly long snippets while at the same time avoiding very short ones. All our results are generated with $L_\mathrm{min} = 20$ and $L_\mathrm{max} = 200$.

Discussion. We derived the above formulas and parameters empirically during our early experiments. We used them for all our results (with the same parameters), and achieved good results. Other choices aimed at addressing the same high-level objectives may work equally well, and may be worth exploring.

3.2. Assigning Scores to Snippets

The above approach generates about a hundred short snippets for a video of a few minutes. To make sense of these snippets, we assign them scores motivated by our design principles (Section 2). We use these scores later in our user interface (Section 4) to help users select the best snippets.

We compute scores by first estimating per-pixel quantities that we later sum over a region of interest $\mathcal{M}$. Formally, to compute a score S on a snippet $[a; b]$, we first define per-pixel values s at each pixel p and sum them over the region $\mathcal{M}$: $S([a; b], \mathcal{M}) = \sum_{p\in\mathcal{M}} s([a; b], p)$. For brevity's sake, we omit the $[a; b]$ operand. This formulation lets users create masks and search for snippets that achieve a desired effect on a specific part of the product. We now describe each scoring function in detail. But first, we explain how to summarize a snippet with a single image which we call a *still*, that is used in the definition of several scoring functions.

Summarizing a Snippet with a Still. Summarizing a snippet with a single *still* image is a useful building block when defining our score functions. We also use the still when producing our final result to create a "base image", representing the static illumination of the scene on top of which the highlights move. We seek an image I_still that shows all the frames at once. A naive solution is to average all the frames, but this generates a bland image in which the highlights have been averaged out. Another option is the per-pixel maximum, but it is sensitive to noise. Instead, we use a per-pixel per-channel soft-max over the snippet:

$$I_\text{still}(p) = \frac{\sum_{i=a}^{b} I_i(p) \exp(\alpha I_i(p))}{\sum_{i=a}^{b} \exp(\alpha I_i(p))} \qquad (1)$$

where the computation is carried out independently for each color channel and α controls the effect: $\alpha = 0$ corresponds to standard averaging and larger values make the result closer to the actual maximum. We use $\alpha = 5$ in all our results.

3.2.1 Color

This function assigns high scores to snippets that reveal the color of the object as opposed to the color of the light reflected on it. Since we use a white light source (a tablet displaying an all-white image), we can use color saturation to differentiate object color from that of highlights. This strategy is similar to that of Boyadzhiev et al. [6], but their approach based on RGB angles, favors dark pixels and requires a correction factor. Instead, we use the RGB distance to the gray diagonal of the RGB cube. We compute this quantity over the still image of the snippet to define the per-pixel score function:

$$s_{\text{color}}(p) = \sqrt{(R_{\text{still}} - \hat{I}_i)^2 + (G_{\text{still}} - \hat{I}_i)^2 + (B_{\text{still}} - \hat{I}_i)^2} \tag{2}$$

where $(R_{\text{still}}, G_{\text{still}}, B_{\text{still}})$ is the color of the pixel p in the still image of the snippet, i.e. I_{still} (p), and $\hat{I} = (R_{\text{still}} + G_{\text{still}} + B_{\text{still}})/3$ is its projection on the black–white axis. This measure does not favor dark pixels because these are all close to each other in the black corner of the RGB cube. In comparison, well-exposed pixels lie in the middle of the cube and can be farther away from the gray diagonal. We observed in our experiments that this metric is effective even with objects that look gray because in practice, they are never perfectly colorless.

3.2.2 Shape and Texture

This score finds snippets that emphasize the shape and texture of the product. The intuition behind our approach is that the structures that repeatedly appear in the captured footage are characteristic of the object (or its texture) while those that are visible in only a few frames are not. Our goal is to rank snippets that reveal these repeated features higher. We build our scoring function upon *structure tensors*. These are a standard image analysis tool and we provide an introduction to them in the supplemental material.

Estimating Structure Similarity. To find whether a snippet shows off characteristic features of the product, we use structure tensors computed with the intensity gradients $\nabla \bar{I}$. For a given snippet, we compare the log-Euclidean sum over the entire video to the tensor computed over its still:

$$s_{\text{struct}}(p) = \text{ntsp}(\exp\Big(\sum_{\text{all } i} \log(\mathbf{T}[\nabla \bar{I}_i])\Big), \mathbf{T}[\nabla \bar{I}_{\text{still}}]) \tag{3}$$

where $\mathbf{T}[\nabla \bar{I}_i]$ is the structure tensor of the intensity gradients, and ntsp indicates the *normalized tensor scalar product* used to compare two tensors (see supplemental material). In order to aggregate the information from several tensors, we work in log-Euclidean space [4] instead of simply adding them. The rationale for this scoring function is that, since the sum is over the entire recorded video which comprises thousands of frames, it captures only the features that appear in many frames. The other occasional features are negligible. Intuitively, this sum is a summary of the main structures of the video, and snippets with similar structure tensor fields show off these characteristic structures well.

3.2.3 Motion

We proceed similarly to score the snippets according to how well the motion visible in them represents the typical motion visible on the objects. We compute the structure tensor $\mathbf{T}[\mathbf{f}]$ of the optical flow $\mathbf{f}$ for each frame. We aggregate it over the entire recorded video and over the snippet only. The rationale is the same as in the previous case: aggregating over the entire video captures only the most characteristic features of the motion and we seek snippets with similar motion features. Formally, the scoring function is:

$$s_{\text{mo}}(p) = \text{ntsp}\Big(\exp\Big(\sum_{\text{all } i} \log(\mathbf{T}[\mathbf{f}_i])\Big), \exp\Big(\sum_{i=a}^{b} \log(\mathbf{T}[\mathbf{f}_i])\Big)\Big) \tag{4}$$

3.2.4 Contours

As we discussed in our design principles, emphasizing object contours with rim highlights is standard practice, e.g., [23]. Our scoring function is based on the observation that under rim lighting, the silhouettes of the product are bright and its center is dark, which approximately looks like the distance function to the object border encoded such that 0 is white and large values are black. We apply the distance transform [19] to the mask to get the distance to the border D which we remap to get $\tilde{D} = 1 - 2\big(D/\max(D)\big)^{\nu}$ which is 1 at the border and -1 at the center, with ν controlling how thick the positive region near the border is, i.e., how thick the rim highlight should be. Although $\nu = 1$ produces acceptable results, we found that it is better to set ν so that the positive and negative regions have approximately the same area. For an object with a circular silhouette, $\nu = \log(2)/(\log(2) - \log(2 - \sqrt{2})) \approx 0.56$ generates equal positive and negative areas. We use this value in the paper.

We then remap the still's intensity so that bright pixels equal 1 and dark ones equal -1, that is: $\tilde{I} = 2\bar{I}_{\text{still}} - 1$. These two quantities define our scoring function:

$$s_{\text{rim}}(p) = \tilde{D}(p)\tilde{I}(p) \tag{5}$$

This score is high for bright pixels near silhouettes and dark pixels near the center, corresponding to rim illumination.

3.2.5 Glittering

Gems and surfaces with fine micro-geometry glitter, and artists often emphasize this effect. We characterize glittering

as the fast variation of the fine-scale details of the image. Formally, we first apply a high-pass filter to each frame's intensity to get a layer $H_i = \bar{I} \otimes (1 - G_\sigma)$ where G_σ is a 2D Gaussian kernel with standard deviation σ. We use $\sigma = 1$ in all our results to capture only the highest-frequency details. Then, we measure the amplitude of the temporal derivative of this layer to define our score:

$$s_{\text{gli}}(p) = |H_{i+1}(p) - H_i(p)| \tag{6}$$

3.2.6 Directional Sweeps

A critical artistic choice is the direction in which the highlights move. The two standard choices are horizontal and vertical sweeps. We provide a scoring function for each by estimating how well the optical flow structure tensor represents the vectors $(1;0)^\mathsf{T}$ and $(0;1)^\mathsf{T}$:

$$s_{\text{hor}}(p) = \sum_{i=a}^{b} \begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{T}[\mathbf{f}_i] \begin{pmatrix} 1 \\ 0 \end{pmatrix} \tag{7a}$$

$$s_{\text{ver}}(p) = \sum_{i=a}^{b} \begin{pmatrix} 0 & 1 \end{pmatrix} \mathbf{T}[\mathbf{f}_i] \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{7b}$$

Summary. Given the user's input footage (a few minutes), we split it into snippets (around 100), and then develop scoring functions that rank the snippets based on various criteria: shape, motion, rim lighting, glitter, and directional sweeps.

4. GUI and Compositing

As the last step in our workflow, the user combines snippets into a final video using our GUI. At startup, we present to the user the familiar preview pane similar to other photo and video editors. Since the video is often dark, we initially show the temporal average frame.

The two main user interactions are to *create a region*, and to *assign a snippet* to a region. Regions are similar to layers in numerous image editing tools in that they are associated with image data (*i.e.*, a snippet), and a blending mask. With the mouse, users can create rectangular or lasso-polygon regions, which we rasterize into an alpha mask with Guided Filtering [10], using the average image as the guide.

We enable the browsing of the available snippets with *faceted search* where the user is given a series of *cascaded lists*, similar to the interface found on many shopping sites. After selecting a region and a primary criterion (*e.g.*, rim lighting), an initial list contains all snippets sorted by the mean score inside the region. The user either chooses one of these snippets, or selects a secondary criterion, in which case our system takes the top 50% of the snippets, sorts their mean scores in the secondary criterion, and presents them in a secondary list. Cascading continues until the user selects

(a) Professionally lit wine bottle from [7] (b) Our criteria applied on different parts (c) Our final result (on a white background)

Figure 1. In the professionally lit photo (a), (1) the *back-lit* body reveals the colors and darkens the contours, (2) the *side highlight* reveals the reflective glass body, and (3) the *shape and texture* on the cap and the logo are emphasized through directional lighting. We achieve these effects (b) by applying the *color* and *vertical motion* criteria to the body, to capture the translucent color, dark contours, and vertical highlight ((1) and (2)), and (3) the *shape & texture* criterion to the cap and logo. Finally, we composite all of these effects over a white background (c).

a snippet. To help the user quickly understand the selected snippet, we provide a summary panel with a temporal soft-max of the snippet, a tone-mapped score image, and the blending mask. The summary panel also has a button to play the snippet in the main preview window (compositing it over the other regions), and a slider to change playback speed.

Compositing. For novice users, traditional alpha compositing can lead to unexpected results when ordered improperly. Therefore, we use the soft-max operator, which is commutative, for spatial compositing as well. For our application, soft-max performs better than other commutative blending operators such as *plus* because our focus is on bright moving lights, which tend to blur under linear combinations. Given a collection of regions, their alpha masks, and corresponding snippets, we premultiply each snippet frame by the alpha mask, apply soft-max across regions, and normalize by alpha.

Sequencing. In lieu of a timeline, we provide a simple but flexible scripting interface to sequence the final video. Users create a series of shots, which are played sequentially with fade to black between them. Each shot is a collection of regions, which are either dynamic (at its selected framerate)

or a still (its temporal soft-max), an optional zoom window, and an optional background. All of our results are created using this system. The supplemental material contains a screen capture of an interactive session.

In our user study (Section 5.1), the more sophisticated users asked for advanced blending modes and nonlinear editing. These features are complementary to our prototype and can be added to make the GUI production quality.

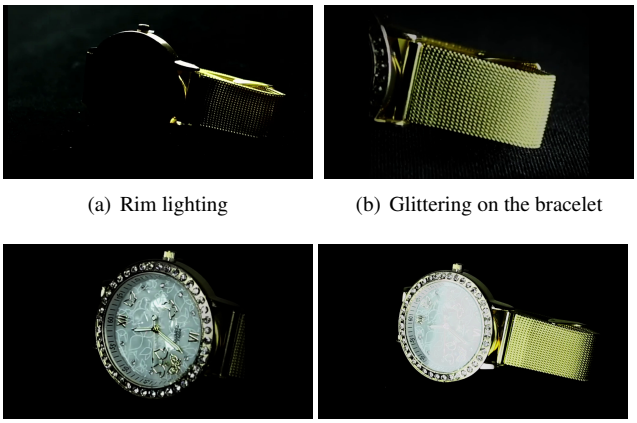
We also experimented with *templates*, i.e., pre-defined scripts to automatically generate videos. Each sequence in a template specifies a type of shot, e.g., rim light or sweep, and we automatically assign the highest-ranked snippet in that category. We enable close-ups by zooming in on the center of the image, which corresponds to the center prior for saliency introduced by [11]. We demonstrate a few examples in the supplemental materials. We envision that trained users can create template libraries that novices can use. While using templates results in more repetitive videos compared to user-created scripts, they can be useful to high-volume sellers who need to generate many videos quickly.

5. Results

We demonstrate the effectiveness of our approach to produce lighting designs on a variety of objects for both still images and short videos. In the supplemental video, we used Adobe Premiere to create a fade-in and fade-out effect between the individual short clips generated by our system and to add music. Those features are orthogonal to our research and they can easily be added to the future versions of our prototype software.

Wine bottle (still). The wine bottle in Figure 1 is a common case in product photography. We demonstrate that our technique allows a quick exploration and combination of classical lighting-design objectives, which are captured by the top few high-ranking snippets sorted according to our criteria. In the supplemental video, we also show a video result.

Golden watch (video). Watches are another common subject in the lighting-design videos we studied. In Figure 2, we show a few frames of a short, 22-second video clip produced with our system. We start by showing the full scene and we play one of the high-ranking snippets that emphasizes the overall shape through rim lighting. Next, we zoom in on a few regions and play snippets that reveal various shape and material properties. We zoom into the case and use the glittering criterion to emphasize the diamonds, composited over a *still* snippet that reveals the texture of the glass. Finally, we zoom out to show a full view of the scene where we *still* a few snippets to get a good base light on top of which we play the highest-ranked snippet that captures a horizontal highlight sweep. This final sweep is often seen in professional videos, where the goal is to attract viewer's attention to the reflective behavior of the glass case. Our *horizontal motions* criterion captures this common goal.



(a) Rim lighting (b) Glittering on the bracelet

(c) Glittering on the diamonds (d) Highlight sweep

Figure 2. A few representative frames of the *Golden Watch* video produced using our system.

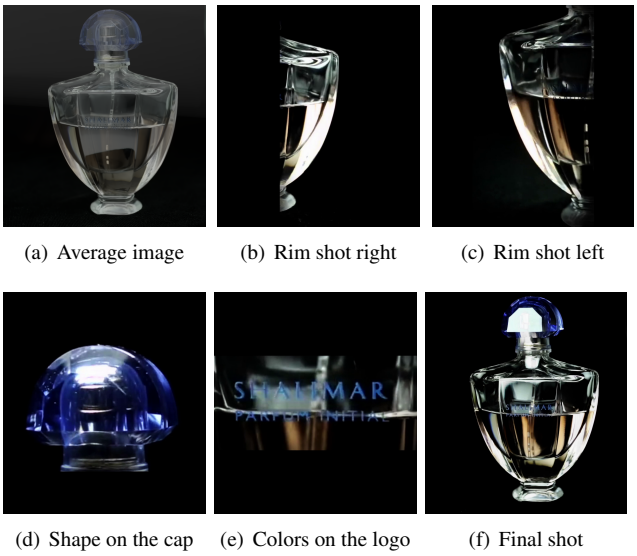
Perfume (video). Perfume bottles made from faceted glass are a challenging case because of the complex interactions with the moving light. In Figure 3, we show a video result on a perfume bottle using our method. To reveal contours, we first play high-ranking rim-light snippets on the left and then right sides of the bottle, which reveal its overall shape. Next, we zoom in on a few regions to emphasize shape (the cap) and material (the logo). Finally, we zoom out and show the *still* image of the highest-ranked snippet that reveals the shape of the body, on top of which we blend an animation of the snippet that highlights the logo.

5.1. Validation

We conducted two small-scale user studies to validate our system.

Study 1: Our Pipeline. The first study was designed to ascertain whether novice users, given only a short tutorial, can use our system to produce good product photography. We asked two novice users who have never seen our system to go through our entire pipeline. They were both tasked with acquiring, analyzing (using our automatic techniques from Section 3), and editing two objects: one chosen by us (*coins*), and the other chosen by them (*tool* and *camera*, respectively). Neither user had much video editing experience, although both have a fair amount of experience with image editing using Photoshop.

The first question both users asked after our tutorial was



(a) Average image (b) Rim shot right (c) Rim shot left

(d) Shape on the cap (e) Colors on the logo (f) Final shot

Figure 3. A few representative frames from shots in our *Perfume* video. We show the full clip in the supplemental video.

"what do I do?" Despite the fact that they chose and acquired a personal item, they were unsure what is good lighting design for product videos. For this study, we did not show the users any professional videos and simply encouraged them to explore the dataset. User A was photographically motivated and spent a significant amount of time creating masks to blend various still lights, before adding a single moving light. User B was more exploratory and browsed around until he found a "favorite". He only applied one light sweep per region before quickly moving on to the next.

The results of the first study demonstrate that amateurs can use our pipeline, although quality depends on "knowing what you want." Novice users need training in lighting design before they can take full advantage of our system. Their results are included in the supplemental material.

Study 2: Snippet Analysis and GUI. The goal of the second study was to assess the quality of our snippet analysis and its usefulness as part of a video production tool. We asked five users to work on the same sequence (*Golden Watch*). Each user was given a single viewing of an actual 40-second watch commercial as a template, a short tutorial on our system, and unlimited time to practice on a training sequence of a leather watch. They were then given 15 minutes to create a video in the spirit of the commercial, and given a short exit survey (see supplemental material).

We found that overall, everyone liked the concept. One user wrote: "I like that it is giving me a tool to emulate professional product shots without having to buy a bunch of gear. Lighting is a big separator between professional and non-professional photographers/videographers." Most users found our snippet analysis "generally helpful, although not completely reliable", and that "The classifiers help, but it's still a pretty big list to sift through." Although half the users mentioned that they found the fully-automatic compositing intuitive, everyone felt that the biggest pain point was the lack of a full-featured nonlinear editor. Users wanted to trim or reverse snippets, and the more advanced users wanted to apply more sophisticated blending.

To summarize, all the users liked the concept of a one-person DIY tool to create professional-looking product videos. They found faceted search of catalogued snippets to be a fast way to find the right effect. However, with regard to the user interface, nearly all users wanted additional features and would have preferred that our tool be part of a nonlinear video editor such as Premiere or iMovie.

5.2. Discussion and Limitations

We describe a user-driven approach meant to help users create compelling lighting-design videography. However, not all criteria configurations are necessarily useful in all situations. For example, if a scene does not have materials with glittering properties, our criterion cannot return a snippet that has the expected behavior. That said, our experiments show that even if one or two criteria do not produce the expected behavior on a given scene, many of the others do.

Our results do not correspond to a physical setup, since our per-region blending does not alter illumination in a physical way. However, it is close to what could be produced using blockers, and our results look plausible. Further, our soft-max operator, which blends frames across snippets in a non-linear manner, is close to what one can get in practice with a longer exposure and also looks plausible.

Our snippet extraction procedure may not always cut where a designer would, as our smoothing scores may not always correspond to what a designer would perceive as a smooth and complete sweep. Nevertheless, we found that our automatically extracted and ranked snippets are useful in quickly directing the designer to a desired point in the video where the effects of various criteria are observed. Sliders to refine the start/end frames of the snippets can easily be added to let a designer further refine snippets.

Finally, we found that our approach is most suitable for novices. Some of our test users had never done video editing before and said that they would not have been able to generate a product video with any other tools. On the other hand, more skilled users asked for more advanced tools typically found in video editing software. This suggests that our approach is a good candidate for inclusion in a standard video editing package to enable novices while assisting advanced users. While such inclusion is beyond the scope of this study, it is a possible direction for future work.

6. Conclusion

The growth of sites like craigslist and eBay is increasing the need for tools that enable easy-to-produce product photographs and videos. Additionally, the increasing ubiquity of electronic billboards and online advertising is increasing the importance of product videography.

We introduce a do-it-yourself lighting design system for product photography and videography. Our pipeline of acquisition, analysis, and compositing lets novice users produce high quality lighting design for products without too much effort. Our simple acquisition pipeline requires no specialized hardware beyond a tablet and a smartphone. We automatically analyze videos to produce snippets that are ranked on various criteria based on whether they reveal shape, texture, motion, glitter, or achieve rim lighting.

Many future avenues of research remain. A more complete production quality UI would improve the user experience. Automatic summarization of the input video could decrease user interaction, except when they want to artistically control the results. Combining this approach with an Arqspin-type product can let us expand the range of effects by combining varying viewpoints and illumination of an object. Exploiting knowledge of the 6DoF tracking of the light and camera to get 3D information could also significantly expand the possibilities.

Acknowledgments. We would like to thank the anonymous reviewers for their constructive comments. We would like to acknowledge our funding agencies NSF CGF 1161645 and funding from Adobe. We thank all the participants of our usability study.

References

[1] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. *ACM TOG*, 2004.

[2] M. Ajmal, M. H. Ashraf, M. Shakir, Y. Abbas, and F. A. Shah. Video summarization: techniques and classification. In *Computer Vision and Graphics*, pages 1–13. Springer, 2012.

[3] D. Akers, F. Losasso, J. Klingner, M. Agrawala, J. Rick, and P. Hanrahan. Conveying shape and features with image-based relighting. In *Proc. of IEEE Visualization*, page 46, 2003.

[4] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache. Log-Euclidean metrics for fast and simple calculus on diffusion tensors. volume 56, pages 411–421, Aug. 2006.

[5] A. Bousseau, E. Chapoulie, R. Ramamoorthi, and M. Agrawala. Optimizing environment maps for material depiction. In *Computer Graphics Forum (Proc. of the Eurographics Symposium on Rendering)*, volume 30, 2011.

[6] I. Boyadzhiev, S. Paris, and K. Bala. User-assisted image compositing for photographic lighting. *ACM TOG*, 2013.

[7] Cravelocal. Example image of lighting design for product photography (http://goo.gl/hzgssr), 2011.

[8] R. Fattal, M. Agrawala, and S. Rusinkiewicz. Multiscale shape and detail enhancement from multi-light image collections. *ACM TOG*, 26(3):51, 2007.

[9] R. W. Fleming, R. O. Dror, and E. H. Adelson. Real-world illumination and the perception of surface reflectance properties. *Journal of Vision*, 3(5), 2003.

[10] K. He, J. Sun, and X. Tang. Guided image filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(6):1397–1409, June 2013.

[11] T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to predict where humans look. In *IEEE International Conference on Computer Vision*, 2009.

[12] K. Karsch, V. Hedau, D. Forsyth, and D. Hoiem. Rendering synthetic objects into legacy photographs. *ACM TOG*, 2011.

[13] J. Kim, P. Marlow, and B. L. Anderson. The perception of gloss depends on highlight congruence with surface shading. *Journal of Vision*, 11(9), 2011.

[14] Z. Liao, N. Joshi, and H. Hoppe. Automated video looping with progressive dynamism. *ACM TOG*, 32(4), 2013.

[15] J. Lopez-Moreno, E. Garces, S. Hadap, E. Reinhard, and D. Gutierrez. Multiple light source estimation in a single image. *Computer Graphics Forum*, 2013.

[16] J. Lopez-Moreno, S. Hadap, E. Reinhard, and D. Gutierrez. Compositing images through light source detection. *Computers and Graphics*, 2010.

[17] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. 1981.

[18] A. Mohan, R. Bailey, J. Waite, J. Tumblin, C. Grimm, and B. Bodenheimer. Tabletop computed lighting for practical digital photography. *IEEE Trans. on Visualization and Computer Graphics*, 13(4), 2007.

[19] S. Natarajan. *Euclidean Distance Transform and Its Applications*. AV Akademikerverlag GmbH & Co. KG., 2010.

[20] S. Rusinkiewicz, M. Burns, and D. DeCarlo. Exaggerated shading for depicting shape and detail. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 25(3), 2006.

[21] A. Schödl, R. Szeliski, D. H. Salesin, and I. Essa. Video textures. In *Proc. of ACM SIGGRAPH*, 2000.

[22] Y. Shih, S. Paris, C. Barnes, W. T. Freeman, and F. Durand. Style transfer for headshot portraits. *ACM TOG*, 33(4), 2014.

[23] M. Srikanth, K. Bala, and F. Durand. Computational rim illumination with aerial robots. In *Proc. of Workshop on Computational Aesthetics*, 2014.

[24] Tissot. Example video of lighting for product videography (https://www.youtube.com/watch?v=bpe88mi0ebq), 2011.

[25] Tissot. Example video of lighting for product videography (https://www.youtube.com/watch?v=lliuhzzkqa8), 2014.

[26] A. Wenger, A. Gardner, C. Tchou, J. Unger, T. Hawkins, and P. Debevec. Performance relighting and reflectance transformation with time-multiplexed illumination. *ACM TOG*, 2005.

[27] H. Winnemöeller, A. Mohan, J. Tumblin, and B. Gooch. Light waving: Estimating light positions from photographs alone. *Computer Graphics Forum*, 24(3), 2005.

[28] B. Xiao, B. Walter, I. Gkioulekas, T. Zickler, E. Adelson, and K. Bala. Looking against the light: How perception of translucency depends on lighting direction. *Journal of Vision*, 14(3), 2014.