Entire Reflective Object Surface Structure Understanding

Qinglin Lu[1]
qinglin.lu@u-bourgogne.fr
Olivier Laligant[1]
olivier.laligant@u-bourgogne.fr
Eric Fauvet[1]
eric.fauvet@u-bourgogne.fr
Anastasia Zakharova[2]
anastasia.zakharova@insa-rouen.fr

[1] University of Burgundy
Le2i UMR 6306 CNRS
12,Rue de la Fonderie,71200,France
[2] INSA Rouen LMI EA3226
Avenue de l'Université,76800,France

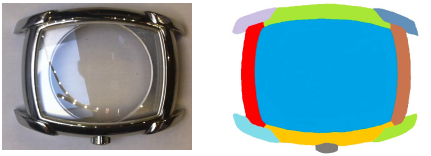
Abstract

Reflection from reflective surface has been a long-standing problem for object recognition, it brings negative effects on object's color, texture and structural information. Because of that, it is not a trivial task to recognize the surface structure affected by the reflection, especially when the object is entirely reflective. Most of the time, reflection is considered as noise. In this paper, we propose a novel method for entire reflective object sub-segmentation by transforming the reflection motion into object surface label. Instead of considering the reflection as noise, our approach takes reflection as an advantage for understanding the surface structure of the entire reflective objects. The experimental results on specular and transparent objects show that the surface structures of the reflective objects can be revealed and the segmentation based on the surface structure outperforms the approaches in literature.

1 Introduction and related work

The *object surface structure* (OSS) describes the geometric distribution of the elementary continuous surfaces of an object (The definition of elementary continuous surface refers to section 3). It is a highly representative local feature. The understanding of the OSS is considered as a building block for solving problems such as object recognition, detection, and classification. For non-reflective objects, the OSS can be easily recognized due to the object contour, texture, and color. However, for the entire reflective objects, the reflective effects make the understanding of OSS extremely complicated. For instance, as referred in figure 1, fig.1a is the original image of an entire reflective object which consists of both specular and transparent surfaces. And, fig.1b is the ground-truth of the manual sub-segmentation according to the OSS. We can see that due to the reflection on the object, the boundaries are barely observable and the OSS is hard to recognize. And seeing through the transparent surface, undesired components inside the object are also visible. Thus, the sub-segmention from fig.1a to fig.1b is not a trivial task. Here, the sub-segmentation allows to differentiate

Pages 116.1-116.11
DOI: https://dx.doi.org/10.5244/C.29.116



(a) (b)

Figure 1: Reflective object structure understanding. (a) original image (b) manually sub-segmented ground-truth image.

object's surfaces in order to have a better understanding of their structure. As a consequence, the objective of this paper is to sub-segment entirely reflective objects by exclusively taking the advantage of reflection.

Many works have been done in dealing with reflection in the image. [7, 14, 19, 20] consider the reflection as noise, they try to remove or reduce it. Also, a few works attempt to use information contained in reflections to extract object features. Savarese and Perona [16, 17] propose an analysis of the relationship between a calibrated scene composed of lines through a point, and the geometry of a curved mirror surface on which the scene is reflected. This analysis is used to measure object surface profile. DelPozo and Savarese [5] use static specular flows features to detect specular surfaces on natural image.

There are also various contributions in video object segmentation. Most existing methods attempt to exploit the temporal and spatial coherence in the image sequance, in which pixels with similar appearance and spatiotemporal continuity are grouped together over a video volume [13, 15, 23]. Felzenszwalb et al. [9] adapts graph-based image segmentation to video segmentation by building the graph in the spatiotemporal volume. Shi and Malik [18] uses nystrom normalized cuts, in which the nystrom approximation is applied to solve the normalized cut problem for spatiotemporal grouping. Grundmann et al. [12] applies hierarchical graph-based approach in segmenting 3D RGBD point clouds by combing depth, color, and temporal information.

Approaches closest to ours investigate in extracting fine-gained attributes for object recognition [2, 6, 8, 10, 11]. Deng and Feifei [6] present an attribute-based framework for describing object in details which is generalized across object categories. Bourdev and Malik [2] use 3D data of human body which is annotated into different body parts to recognize the pose. Tsogkas et al. [22] proposes a method for understanding objects in detail by studying the relation between part detection and attribute prediction. It diagnoses the performance of classifier that pool information from different parts of an object. As we are working on reflective and transparent objects, the difference of data type makes those methods not comparable although all of us mean to sub-segment the objects.

In this paper, our proposed approach extracts reflection motion features in the image sequence as spatiotemporal information, then sub-segment object by taking these features in order to understand the OSS. The setup of our method is straightforward, the positions of camera and object are fixed, however the light source is moving around the object in order to produce *reflection particles* (RP) on the object surface. While the RP are moving on the object surfaces, their positions, directions, and velocities are extracted in each frame as reflection motion features. These features are matched in all the frames for tracking RP in the whole sequence. We assume that the RP move smoothly along an elementary continuous surface and irregularly while passing from one surface to another. Thus, we break tracking when the motion features are irregularly compare to that in the previous frames. This

guarantees to keep the trajectory of a moving RP stay on one elementary continuous surface. Later, one elementary continuous surface is segmented by employing flood fill method [21] which takes the positions in the trajectory as seeds. As this process iteratively covers all the trajectories, different surfaces of the object could be respectively labeled.

Our primary contributions are: (1) An effective sub-segmentation method for the reflective surface structure understanding (on both specular and transparent surfaces). (2) Instead of removing reflection, we take it into account as information for sub-segmentation and prove that reflection can be advantage for object recognition. (3) The use of reflection motion features as spatiotemporal coherence for video segmentation and fine-attributes for OSS understanding.

2 Motion Estimation of Reflection

Our goal is to transform the motion of reflections into useful information that can help to segment the different elementary continuous surfaces of an object. According to that, we firstly extract RP motion features, then track them in video frames.

2.1 Reflection motion features extraction

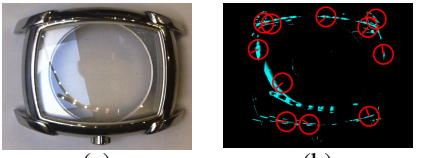
Since the object and camera are fixed, in the video, significant movements are produced by reflections due to the movement of the light source. We employ the motion history image [1, 4] (*MHI*) to extract RP. The *MHI* $H_\tau(x,y,t)$ can be computed from an update function $\Psi_\tau(x,y,t)$:

$$H_\tau(x,y,t) = \begin{cases} \tau & if \quad \Psi_\tau(x,y,t) = 1 \\ max(0, H_\tau(x,y,t-1) - \delta) & if \quad \Psi_\tau(x,y,t) = 0 \end{cases} \tag{1}$$

Here $\Psi_\tau(x,y,t)$ denotes motion at position (x,y) in t-th frame, the duration τ decides the temporal extent of the movement, and δ is the decay parameter. This computation leads to a static scalar valued image where the more recently moving pixels are brighter. Then the moving direction can be efficiently calculated by convolution with separable Sobel filters in the X and Y directions yielding the spatial derivatives: $F_x(x,y)$ and $F_y(x,y)$. So, the gradient orientation (ϕ) of the pixel is then: $\phi = \arctan \frac{F_y(x,y)}{F_x(x,y)}$.

These gradient vectors will point orthogonally to moving object boundaries at each step in the *MHI*. It gives us a normal optical flow representation. After that, a downward stepping flood fill [21] is used to label motion regions connected to the current *MHI*. This computation collects the neighbor pixels which have similar motions as a connected RP. From the frame at time t, we extract the n moving RP as $C_i^t = \{8\text{-}connected\ pixels\ of\ the\ same\ motion\}$ where $i \in [1:n]$.

From each RP (C_i^t), a motion feature vector $f(C_i^t)$ is extracted where $f(C_i^t) = \{d_i^t, p_i^t, v_i^t\}$. d_i^t, p_i^t, and v_i^t present the direction, the position, and the velocity of the RP, respectively. The features extraction is illustrated as follow: d_i^t is obtained by taking the average direction of all the pixels in C_i^t; p_i^t is the center of a bounding box that contains C_i^t; v_i^t is the euclidean distance between positions of two continuous frames ($p_i^t - p_i^{t-1}$), note that v_i^t is computed during tracking the corresponding C_i^t.



(a) (b)

Figure 2: (a) Original frame; (b) motion history image of current frame. White pixels represent moving reflection particles. Red clocks represent moving directions of correspondent reflection particles.

2.2 Reflection Particle Tracking

The tracking of RP suffers from several problems: the high frequency of appearance and disappearance of the RP, the shape evolution of the RP, as well as multiple reference RP need to be tracked in the same time. Our tracker is composed by an iterative matching computation. The tracker is initialized for each detection, the state of a reference RP (C_i^t) is presented as $S(C_i^t) = \{p_i^t, d_i^t, v_i^t\}$. The state transition density is defined as follows:

$$p_i^t = p_i^{t-1} + v_i^{t-1} \times 1, \qquad v_i^t = v_i^{t-\Delta t}. \tag{2}$$

The sampling processes a predictive circle window with the radius of δ and the center at the position predicted by equation 2. It is due to the RP motion features have already been extracted in each frame. Instead of sampling candidate RP (note as cc_j^t with its feature $f(cc_j^t) = \{dc_i^t, pc_i^t, vc_i^t\}$) with a weight which costs computationally expensive, a predictive sampling window is employed. Then each reference RP and candidate RP pair in the predictive window is scored by the difference of the moving direction:

$$err_d^c(c_i^t, cc_j^t) = (d_i^t - dc_j^t)^2, \tag{3}$$

and the $Argmin\left\{err_d^c(c_i^t, cc_j^t\right\}$ is computed to find the best match. Here we also present a threshold parameter β to break current reference RP tracking when the RP moving direction hugely changes. In our experiments, the value of β is set to 30. This tracking phase guarantees to keep all the associated RP on the same surface.

During tracking RP in frames, positions of all tracking results are saved as the moving trajectory. The trajectory of C_i is denoted as $T(C_i) = \left\{p_i^1, p_i^2, ...p_i^t, \right\}$. One trajectory is considered as one label for a continuous surface on the object. As the RP could go through one surface in different directions, we save trajectories respectively for each direction. In this case, it ensures that one trajectory labels only one surface. On the other hand, some trajectories are labeling the same surface. In figure 3, where one color presents one trajectory of moving reflection, image 3.1 contains 15 longest trajectories, image 3.2 contains all the trajectories.

3 Elementary Continuous Surfaces Segmentation

For convenience, we first introduce elementary continuous surface. It is defined according to the variation of γ of the object surfaces, γ being the difference between the neighboring normal as a given direction. The distance between two neighbor points on the object surface

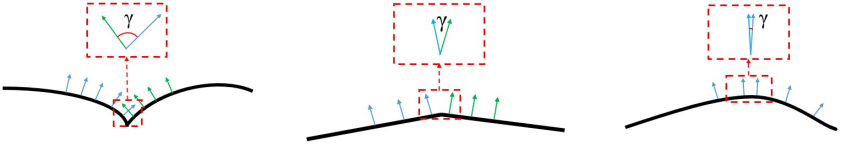
(a) (b)

Figure 3: Reflection moving trajectories. (a) fifteen longest trajectories. (b) all the trajectories.

is defined as 1 mm. Then whether a surface is an elementary continuous surface is verified by $D(\gamma)$, where

$$D(\gamma) = \begin{cases} true & if \quad \gamma > \psi \\ false & if \quad \gamma < \psi \end{cases} \tag{4}$$

Along a surface, if all the corresponding γ below the threshold parameter ψ, the surface is considered as an elementary continuous surface, otherwise it is not. Here ψ denotes the limit of a surface normal variance which is not visible in the image. After experiments on various objects, ψ is set to 2.2 degrees by experience. As shown in figure 4, fig.4a and fig.4b are discontinuous surfaces since their γ are beyond the threshold parameter ψ, while fig.4c is an elementary continuous surface as its low variation of γ along the whole surface.



(a) (b) (c)

Figure 4: (a)(b) discontinuous surfaces (c) elementary continuous surface.

Segmentation of the elementary continuous surfaces is to describe the surface structure of the object. As several trajectories might label the same surface, an iterative flood fill method [21] is applied to merge the segmentation results of different trajectories on the same surface. The seeds which need to be flood filled are systematic sampled positions with a skip of 5 in the trajectory. As the positions are interspersed on one surface, flood fill all these seeds (positions) with a same color votes for one continuous surface, even though in the original image, this surface is displayed as different surfaces due to reflections. The flood fill method which we used during the segmentation is 8-connectivity. The pixel value at (x,y) is considered to belong to the labeling domain if:

$$I(x',y') - d_l < I(x,y) < I(x',y') + d_h, \tag{5}$$

where d_l and d_h present maximum lower/upper brightness difference between the current observed pixel and one of its neighbors belonging to the surface, respectively. Algorithm of the segmentation process is illustrated in the algorithm 1.

Since the trajectories do not have the same length, we put trajectories in an order by increasing lengths and systematic sample the positions by a skip of 5, flood fill from sampled seeds in shorter trajectories to the sampled seeds in longer trajectories. In this case, the parts of one surface which has already been labeled by seeds in shorter trajectories could be merged into other parts of this surface by the labeling of seeds in longer trajectories. As

Algorithm 1 Segmentation process

1. Trajectory sampling
   (a) Sort trajectories by size in increasing order
   (b) Systematic sampling of each trajectory with a skip of 5
2. Segmentation
   (a) Update labeling color to the color of $T(C_i)$
   (b) Flood fill all $p_i^t \in T(C_i)$ with current labeling color
3. Morphology component regrouping
   (a) Update current labeling color to the color of $T(C_j)$
   (b) Regroup and fill all the components passed by $T(C_j)$ with current labeling color ($i < j$)
4. Final processing
   (a) Fill holes which are surrounded by segmented regions with the surrounding color

the reflection on the surface is highly variable, the segmentation phase might not cover the whole surfaces. In consequence, the final processing fills the holes which are surrounded by segmented regions with the surrounding color.

4 Results and Evaluation

The experiments are conducted in using the camera with the resolution of 5 Mpx. A LED grow light is used to produce reflections on the object. Note that the light source is consisted by multiple light dots and they can be any shape, here we use round ones. For the outdoor experiments, two projectors are used. The number of acquired frames is depending on the complexity of the object surfaces and the number of light sources. In order to keep the number of acquired image not expanded, our LED grow light contains 30 light spots.

As the considered objects are reflective and/or transparent, the images contain many high-variability regions. Three of the comparison segmentation methods are graph based [9, 13]. They are based on k nearest neighbors, adjacent, and hierarchical graph, respectively. The graph-based methods are chosen since they have the ability to preserve detail in low-variability image regions while ignoring detail in high-variability regions. The forth comparison method is EM segmentation [3]. It is a pixel clustering method in a joint feature space. It segments the image with the information from different aspects (color-texture-position). Over 20 objects have been experimented on, 6 of them are shown in the figure 6. Due to the similarity of the three graph-based results and the lack of space, only KNN graph-based results are illustrated in figure 6. The first two objects, light cover and ball have completely specular surfaces, the third object scotch is transparent, and the other three objects contain both specular and transparent surfaces. From the results, we can see that graph-based methods work reasonably in segmenting the object, but about the sub-segmentation of the object surfaces, it does not work meaningfully. EM segmentation preserves very well the contour of the objects but also the contour of the reflection that yields the poor sub-segmentation performance. Conspicuously, the results obtained by our method are more accurate. In consequences of a high sub-segmentation performance, the OSS is well presented.

4.1 Quantitative Evaluation of our segmentation results

The purpose of our object surfaces segmentation is to understand the structure of the reflective objects. Therefore, to evaluate our proposed method, we manually labeled all the elementary continuous surfaces of the object to generate the ground-truth image as reference. Then we verify the segmentation performance with a pixel-wise evaluation.

4.1.1 Evaluation in details

To evaluate our proposed method in details, we calculate true positives (TP), false positives (FP), false negatives (FN), precision and recall for each surface, which are computed as follow:

$$TP = \frac{NTP}{PG}, \quad FP = \frac{NFP}{PD}, \quad precision = \frac{NTP}{NTP+NFP}, \quad recall = \frac{NTP}{NTP+NFN}, \tag{6}$$

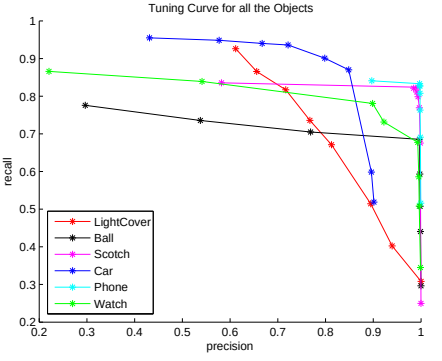
Where NTP, NFP, NFN stand for the number of the true positive pixels, false positive pixels and false negative pixels, respectively. PD, PG, ND, NG stand for number of positives detected, number of positives in ground-truth mask, number of negatives detected and number of negatives in ground-truth. After computing precision and recall for each surface, a weighted combination of evaluations on each surface is proposed to verify the entire performance for a whole object. Total pixel numbers of the object in the ground-truth N is computed as: $N = \sum_{i=1}^{n} PG(i)$, where n is the number of surfaces. Then a weight w_i is defined by the percentage of the pixel number of current surface on that of the whole object, shown in equation 7, where i is surface index.

$$w_i = \frac{PD(i)}{N}, \tag{7}$$

with the weights of each surface, the precision ($precision_o$) and recall ($recall_o$) of the object can be computed as follow:

$$precision_o = \sum_{i=1}^{n} precision_i \times w_i; \quad recall_o = \sum_{i=1}^{n} recall_i \times w_i; \tag{8}$$

Then, we generate the *receiver operating characteristic* (ROC curves) for objects in the experiment by varying the parameters d_l and d_h of the flood fill method. we use 5 different values for $d_l \in [1.5, 2.5, 3.5, 4.5, 5.5]$ and 3 different values for $d_h \in [6.5, 7.5, 8.5]$. From the ROC curves, we can see that for Scotch, Ball and Phone, the precision values keep very high at the beginning and suddenly go down during the raising of recall values. This is due to the fact that these objects all have two surfaces. Within the change of parameters of flood fill method, the labeling color of one surface overfills the other surface. Then the sudden overfilling makes precision value suddenly drop down. For the other objects, as they have approximately ten surfaces, the curves are more smooth. For all the indoor experiments (except the one of the car), the precision values reach 99% and recall values are more than 78%. For the outdoor experiment on the car, under a nature environment without controlling other illumination condition except our light source, the precision value reach to 90.1% and the recall value is 93.5%. These results illustrate the robustness of our segmentation method in OSS understanding under different experiment conditions and over various objects.



(a)

Figure 5: ROC curve for the objects. All the curves were generated in using d_l from 1.5 to 5.5, d_h from 6.5 to 9.5. Objects have more surfaces have smoother curves. (better in color)

4.1.2 Comparison with other works

To evaluate the performance of our approach against the segmentation methods in literature (KNN and adjacent graph-based, EM), we employ the f_{score} (FS) as an criterion for each object. FS is computed from the corresponding precision and recall values as follow:

$$FS = 2 \times \frac{precision \times recall}{precision + recall}. \tag{9}$$

FS is a the harmonic mean of precision and recall which globally evaluates the segmentation performance. Therefore, we choose FS as the criterion of segmentation performance evaluation in order to compare our proposed method with the state-of-the-art approaches. In table 1, we compare our proposed method to 4 segmentation methods. The illustrated results are based on the highest obtained FS. We can see that our FS of object 'LightCover' is 0.76, much lower than our results of other objects. It is because that the surfaces of this object are concave, moving reflection vanish extremely quick even though the surfaces are smooth, moving trajectories are split into smaller trajectories. On the other hand, FS of 'Ball' is also only 0.84, because of on this object, pixel values vary a lot in very small regions. Besides that, in the experiment of object 'Phone', pixel values also vary a lot but not rendezvous in small regions because that the surface structure is less complicated. Thus, the final processing of our method can fill the holes and yield the value of FS to 0.91. We would like to emphasize that, in dealing with reflective and transparent objects, our method outperforms significantly (at least 6% higher) the state-of-the-art methods. To point out, only the proposed method and hierarchical graph-based method [13] take advantage of temporal information while the other methods use static data.

5 Conclusion and Perspectives

We presented a segmentation method based on reflection motion features in order to deal with reflective and transparent objects. Our method helps to understand the surface structure of the objects. Due to a very simple constraints involved during the process, our method can be widely used in the industry as object recognition, tracking, and retrieving. The results show

F_{score}	LightCover	Ball	Scotch	Car	Phone	Watch
KNN graph [9]	0.55	0.38	0.48	0.73	0.51	0.74
ajdacent graph [9]	0.48	0.34	0.54	0.66	0.48	0.75
EM [3]	0.17	0.41	0.46	0.54	0.79	0.47
Hierarchical graph [13]	0.46	0.32	0.39	0.72	0.43	0.44
our method	**0.76**	**0.84**	**0.89**	**0.86**	**0.91**	**0.84**

Table 1: Best F_{score} of the objects.

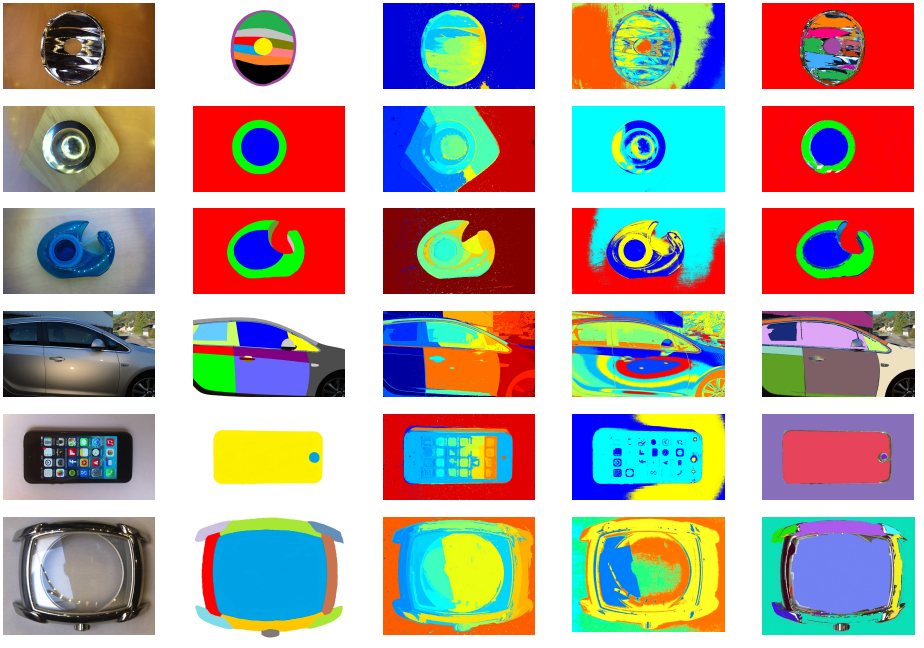


Figure 6: First column: original images. Second column: ground-truth segmentation. Third column: k nearest neighborhood graph-based segmentation [9]. Forth column: EM segmentation [3]. Last column: Segmentation by our proposed method. (better see in color)

that the reflection motion features can be used as a robust signature for labeling continuous surfaces on reflective and transparent objects. In comparison with conventional segmentation approaches, our method can overcome the issues raised by reflective and transparent objects, leading to higher performances in term of accuracy and robustness. This efficiency has been proved through multiple assessments over various objects and under different type of illumination conditions (indoor and outdoor). This series of test highlight the advantage given by our approach against the state-of-the-art methods. More importantly, instead of removing and reducing reflections, taking its advantage is pioneering work in a new direction.

Regarding future work, since the shape of reflection change because of the movement of light source, and hence its reflection on the object surface. We are eager to explore the evolution of reflection shape and extract additional reflection motion features. Furthermore, moving objects instead of moving light source and 3D reconstruction from reflection motion features are also the subjects of our future investigations.

Acknowledgments

This work is supported by the company *ALITHEON* and french national research and technology association (*ANRT*). We are extremely grateful to members of laboratory Le2i for day to day help and collaboration.

References

[1] Md. Atiqur Rahman Ahad, J. K. Tan, H. Kim, and S. Ishikawa. Motion history image: Its variants and applications. *Machine Vision and Applications*, 23:255–281, 2012.

[2] L. Bourdev and J. Malik. Body part detectors trained using 3d human pose annotations. *ICCV*, 2009.

[3] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:1026–1038, 1999.

[4] J. Davis. Hierarchical motion history images for recognizing human motion. *IEEE workshop DREV*, 2001.

[5] A. Delpozo and S. Savarese. Detecting specular surfaces on natural images. *CVPR*, 2007.

[6] J. Deng and L. Feifei. Fine-grained crowdsourcing for fine-grained recognition. *CVPR*, 2013.

[7] M. D'Zmura and P. Lennie. Mechanism of color constancy. *JOSA*, 3:1162–1172, 1986.

[8] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. *CVPR*, 2009.

[9] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59, 2004.

[10] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 32, 2010.

[11] V. Ferrari and A. Zisserman. Learning visual attributes. *Advances in Neural Information Processing Systems*, 2007.

[12] M. Grundmann, V. Kwatra, M. Han, and I. Essa. efficient hierarchical graph-based segmentation of rbgd videos. *CVPR*, 2010.

[13] M. Grundmann, V. Kwatra, M. Han, and I. Essa. Efficient hierarchical graph-based video segmentation. *CVPR*, 2010.

[14] X. Guo, X. Chao, and Y. Ma. Robust separation of reflection from multiple images. *CVPR*, 2014.

[15] S. Paris and F. Durand. A topological approach to hierarchical segmentation using mean shift. *CVPR*, 2007.

[16] S. Savarese and P. Perona. Local analysis for 3d reconstruction of specular surfaces. *CVPR*, 2001.

[17] S. Savarese and P. Perona. Local analysis for 3d reconstruction of specular surfacesâĂŞpart ii. *ECCV*, 2002.

[18] J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22:888–905, 2000.

[19] R.T. Tan and K. Ikeuchi. Reflection components decomposition of textured surfaces using linear basis functions. *CVPR*, 2005.

[20] R.T. Tan and K. Ikeuchi. Separating reflection components of textured surfaces using a single image. *PAMI*, 25:178–193, 2005.

[21] A. Treuenfels. An efficient flood visit algorithm. *C/C++ Users Journal*, 12, 1994.

[22] A. Vedaldi, S. Mahendran, and S. Tsogkas. Understanding objects in detail with fine-gained attributes. *CVPR*, 2014.

[23] C. Xu and J.J. Corso. Evaluation of seper-voxel methods for early video processing. *CVPR*, 2012.