Object Extraction via Stereo Matching and GrabCut

Hyeongseok Son
POSTECH
sonhs@postech.ac.kr

Jaehwan Lee
POSTECH
jhlee525@postech.ac.kr

Seungyong Lee
POSTECH
leesy@postech.ac.kr

Abstract

We propose a disparity refinement algorithm for stereo matching that can be used for foreground object extraction. Depth information obtained by stereo matching is useful for separating objects but usually has less accuracy in object boundaries, due to occlusions between different views. This artifact makes it hard to extract correct object boundaries from images. Our algorithm refines the disparity values around object boundaries by aligning the input color image and the disparity map. We also incorporate the obtained depth information into the GrabCut method for foreground/background segmentation. Experimental results show that accurate boundaries between foreground and background objects can be obtained by our approach.

1. Introduction

Recently there have been many attempts using depth information to enhance results of object-handling problems in computer vision, e.g., scene layout estimation and object pose estimation. For these purposes, it is currently common to use active-type depth-acquisition devices, especially based on infra-red cameras. This type of depth acquisition devices, however, have limitation in processing general scenes, including outdoor scenes and black objects.

One possible solution to avoid this limitation is using a passive-type depth acquisition method, like stereo matching. However, there exists geometric ambiguity caused by occluded regions in object boundaries that prevents the results of stereo matching from being directly utilized for object handling. In this paper, we propose a depth refinement method to reduce this problem, enabling stereo matching to provide depth information adequate for foreground object detection. We demonstrate the effectiveness of our method by applying the resulting depth information to GrabCut algorithm [4] for foreground object extraction. Our method is simple and fast, and can be applied to various stereo matching algorithms while not degrading the accuracy of the baseline algorithms.

2. Algorithm

Our algorithm uses the disparity image obtained by a stereo matching algorithm as the input. We reduce the ambiguity in occlusion regions by aligning object edges between the original color image and the disparity image. Our algorithm replaces an incorrect disparity of a pixel near an object boundary by the disparity of a pixel far from the object boundary, which can be considered as a *safe pixel*.

2.1. Disparity candidate generation

Our algorithm uses a new filter, which is inspired by patch-shift [1], for bringing the depth information of pixels far from an edge to pixels near the edge. Although there are many stable disparities in a region far from an edge, all the pixels do not have ideal disparities for a pixel near the edge. We modify patch-shift for our setting because original patch-shift does not consider that the edges could be misaligned between the input and guidance images.

Let p denote a pixel that we want to correct and p' be a pixel in the patch Ω_p centered at p. For finding p^*, the pixel expected to have the most similar value to the ideal disparity of p, our algorithm first gathers candidates p' that satisfy the following conditions: (1) Pixel far away from the object edge, (2) Pixel contained in the same object as p, (3) Pixel having high confidence for its disparity.

First, in order to distinguish the boundary and interior of an object, we use modified RTV (mRTV) proposed in [1]. mRTV of a pixel in a color image is computed by

$$mRTV\left(p\right) = \triangle\left(\Omega_p\right) \frac{\max\left|\left(\partial I\right)_r\right|}{\sum_{r\in\Omega_p}\left|\left(\partial I\right)_r + \epsilon\right|}. \quad (1)$$

A low mRTV value means that pixel p is far from structural edges. Then pixels that satisfy the following condition will be candidates.

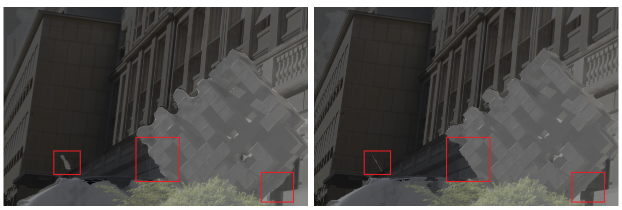
$$mRTV\left(p\right) \le T_{mRTV} \quad (2)$$

Second, we assume two pixels in the same object have similar disparities. We check whether two pixels belong to the same object using color similarity, represented by

$$\left\|I_{left}\left(p\right) - I_{left}\left(p'\right)\right\| < T_{col} \quad (3)$$

(a) Input foreground boundary (b) Using RGB only (c) SGBM (d) Our method

Figure 1: Results of GrabCut using different input values. (b) uses RGB values, while (c) and (d) use the depth information and chromaticity. The result in (b) includes an undesirable region because of color similarity. The result in (c) has been enhanced, but contains less accurate object boundaries. Our result in (d) shows clear edges even when colors are similar.



(a) SGBM (b) Our method

Figure 2: Overlap images of the disparity map and color image. (a) shows that the disparity map of SGBM is misaligned with the color image. Our result in (b) shows reduced occlusion errors and aligned edges.

Third, to avoid bringing a disparity from an occlusion region, we use a cross-checking confidence measure proposed in [2], which is known to be robust for occlusion error checking. A disparity of a pixel in the left image should be same as that of the corresponding pixel in the right image, which is represented by

$$d_{right}\left(d_{left}\left(p'\right)\right) = p' \quad (4)$$

The final candidates Ω'_p are the pixels in Ω_p which satisfy all three conditions in Eqs. 2, 3, and 4.

2.2. Disparity selection

We determine the final disparity using the disparities of the candidate pixels obtained in Sec. 2.2. We expect that most of the candidates have correct disparities, and select the median as the ideal disparity of p.

For a given disparity image, we replace the disparity of every pixel p by the ideal disparity obtained from the candidate pixels. As a result, disparities around object edges are refined, while interior pixels have little changes.

3. Result

The result of our disparity refinement algorithm is shown in Fig. 2. We use the SGBM algorithm [3] with simple hole-filling to obtain the initial disparity image. Our algorithm is highly parallelizable using CUDA, and the refinement step takes about 0.1 second for an 800×600 image in our experimental environment which uses i7 4770 CPU and GTX 780 Ti GPU.

To apply the resulting depth information to the GrabCut method [4], we convert the given input RGB image into lab space and replace RGB channels by the disparity and chromaticity (ab) values. As shown in Fig. 1, Grabcut with the refined disparity values could distinguish even the ambiguous regions where colors of two objects are similar.

4. Conclusion

We proposed an algorithm to enhance the quality of a disparity map in edge regions. We demonstrated that the enhancement can improves the performance of GrabCut for foreground object segmentation. Our algorithm can be combined with an arbitrary stereo matching algorithm, and could be extended for handling depth maps obtained by other methods, such as using MS Kinect.

Acknowledgements

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (R0126-15-1078).

References

[1] H. Cho, H. Lee, H. Kang, and S. Lee. Bilateral texture filtering. *ACM Transactions on Graphics*, 33(4):1–8, 2014.

[2] G. Egnal and R. P. Wildes. Detecting binocular half-occlusions: Empirical comparisons of five approaches. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1127–1133, 2002.

[3] H. Hirschmüller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2008.

[4] C. Rother, V. Kolmogorov, and A. Blake. ”GrabCut” - Interactive Foreground Extraction using Iterated Graph Cuts. *ACM Transactions on Graphics*, 23(3):309, aug 2004.