DISPARITY CONTOUR GROUPING FOR MULTI-OBJECT SEGMENTATION IN DYNAMICALLY TEXTURED SCENES[*]

Wei Sun
Hewlett-Packard Laboratories, 1501 Page Mill Rd. ms 1181, Palo Alto, California 94304-1126, USA

Stephen P. Spackman
TimeSpring Software Corporation, 2000 Peel St. Suite 888, Montreal, Quebec H3A 2W5, Canada

Keywords: Multi-object segmentation, stereo matching, background model, disparity verification, disparity contours.

Abstract: A fast multi-object segmentation algorithm based on disparity contour grouping is described. It segments multiple objects at a wide range of depths from backgrounds of known geometry in a manner insensitive to changing lighting and the dynamic texture of, for example, display surfaces. Not relying on stereo reconstruction or prior knowledge of foreground objects, it is fast enough on commodity hardware for some real-time applications. Experimental results demonstrate its ability to extract object contour from a complex scene and distinguish multiple objects even when they are close together or partially occluded.

1 INTRODUCTION

Extracting moving objects from a static or dynamic scene is useful in a wide variety of applications. The particular case of isolating and distinguishing multiple objects despite rapid changes in illumination and texture is particularly interesting for augmented reality, immersive telepresence, and the entertainment and film industry, where projected moving backgrounds are often present. An example is shown in Fig. 1, where local and remote users interact with each other and with virtual objects in a virtual world. To successfully immerse users into this synthetic environment, it is necessary for the system to separate them visually from their actual physical surroundings.

Single view background subtraction (Wren et al., 1997; Stauffer and Grimson, 1999; Toyama et al., 1999; Oliver et al., 2000; Rittscher et al., 2000; Cucchiara et al., 2003) compares each image to a reference and labels pixels as background or foreground based on a statistical model. Despite adaptability to slow changes in lighting, texture, geometry and shadow, these methods all assume background change to be much less dynamic than foreground.

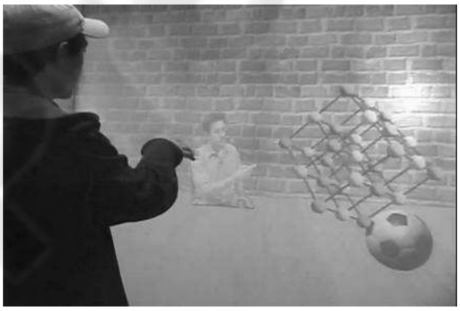


Figure 1: Example of an augmented reality environment.

Layered motion segmentation (Wang and Adelson, 1993; Jepson and Black, 1993; Ayer and Sawhney, 1995; Weiss and Adelson, 1996) decomposes image sequences into sets of overlapping layers ordered by depth, each described by a smooth optical flow field. Discontinuities in the description are attributed to moving occlusions, resulting in a 2.5D scene representation. Unfortunately, the computation of optical flow is time-consuming, and these methods cannot distinguish a real scene from a video.

Integrating multiple views is a natural alternative for tackling dynamic environments, with added benefits in handling occlusion. Some systems work from 3D reconstruction to object segmentation and tracking (Narayanan et al., 1998), and others combine segmentation with stereo matching (Torr et al., 2001; Lin and Tomasi, 2004). Sadly, frame-by-frame stereo reconstruction is also slow and so far unsuited to real-

[*] This research was carried out at the Centre for Intelligent Machines, McGill University, Montreal, Quebec, CANADA.

Sun W. and P. Spackman S. (2007).
DISPARITY CONTOUR GROUPING FOR MULTI-OBJECT SEGMENTATION IN DYNAMICALLY TEXTURED SCENES.
In *Proceedings of the Second International Conference on Computer Vision Theory and Applications - IU/MTSV*, pages 347-354

time use. Moreover, the uniform or repetitive textures common in indoor scenes and video-augmented spaces constitute worst-case inputs for stereo matching algorithms (Kolmogorov and Zabih, 2002; Sun et al., 2003), often leading to disappointing results.

Attempts have been made to use stereo while reducing computational cost. One of them combines stereo with background subtraction and suggests disparity verification for segmentation under rapid illumination change (Ivanov et al., 2000). Using three cameras on wide baselines, the method constructs offline disparity mappings for the background images, and at runtime separates foreground from background by matching pixels corresponding in the mappings, thus avoiding slow disparity search. Unfortunately, the wide baseline setup, despite its effectiveness in extracting the entire foreground area, has difficulty fusing multiple views of a target, which is essential for tracking multiple moving objects. This weakness, in the long run, also limits the method's adaptability to background geometry changes.

Another approach increases speed by decreasing the number of disparity layers in stereo matching and proposes layered dynamic programming and layered graph cut for foreground/background separation (Kolmogorov et al., 2005). Although tolerance of background motion has been demonstrated, published results show only cases with a substantial depth difference between background and foreground, with foreground objects very close to the camera. This is a strong limitation for many real-world applications.

Both fast stereo approaches, however, stop at bilayer pixel labelling, and do not attempt to distinguish multiple objects. The additional processing required for accurate object location would be extensive.

This paper presents a new background subtraction based stereo segmentation system that can isolate and distinguish multiple objects in the presence of highly dynamic lighting and background texture. In addition to the advantage of bypassing full stereo reconstruction and achieving fast performance, it adopts a small baseline setup and extracts *disparity contours*. These directly provide object boundaries, and also allow fast, incremental disparity adjustment for objects at different depths. The method has the potential to provide support for 2D and 3D object tracking and background geometry update.

2 DISPARITY CONTOURS

Before we explain our system in detail, we introduce the disparity contours resulting from small baseline stereo background subtraction. We then show how to estimate foreground disparity and verify object hypotheses based on extracted disparity contours.

2.1 Background Hypothesis Falsification (BHF)

The 3D layout of a background is represented by a **background disparity map** (BDM) describing the relative displacement, or **disparity**, of pixels corresponding to the same background point in each camera view. The images are undistorted and rectified so that pairs of conjugate epipolar lines become colinear and horizontal (Fusiello et al., 2000). This brings the pixels originating from a scene point s to a common scanline, falling at $(\mathbf{x}_\mathrm{L}(s), \mathbf{y}(s))$ in V_L, the left view, and $(\mathbf{x}_\mathrm{R}(s), \mathbf{y}(s))$ in V_R. We call the difference

$$\mathbf{x}_\mathrm{L}(s) - \mathbf{x}_\mathrm{R}(s) = \mathbf{d}_\mathrm{B}(s), \tag{1}$$

increasing with proximity to the camera, the **background disparity** at s, and define the BDM to be

$$\mathrm{BDM} = \{\langle \mathbf{x}_\mathrm{L}(s), \mathbf{x}_\mathrm{R}(s), \mathbf{y}(s)\rangle\}, \tag{2}$$

where s ranges over background scene points visible to either camera and in their common field of view.

Given the BDM for two cameras, each new pair of captured images are hypothesized to be of background alone, and a **view difference map** (VDM) computed from the stored correspondences by block matching, using a vertical stripe window to maintain contour widths and aggregate neighborhood support:

$$\begin{aligned} &\mathrm{VDM}_\mathrm{BHF}(x_\mathrm{L}, x_\mathrm{R}, y) = \\ &\quad \sum_{u,v} \left|\mathrm{V}_\mathrm{L}(x_\mathrm{L}+u, y+v) - \mathrm{V}_\mathrm{R}(x_\mathrm{R}+u, y+v)\right|, \\ &\quad \text{where } y_\mathrm{L} = y_\mathrm{R} = y \text{ and } \langle x_\mathrm{L}, x_\mathrm{R}, y\rangle \in \mathrm{BDM}. \end{aligned} \tag{3}$$

If the images are well synchronized, this operation cancels instantaneous background texture.

Thus, ideally, $\mathrm{VDM}_\mathrm{BHF}(x_\mathrm{L}, x_\mathrm{R}, y) = 0$ where a scene point s is truly part of the background, but is larger if either of the pixels $\mathrm{V}_\mathrm{L}(x_\mathrm{L}, y)$ and $\mathrm{V}_\mathrm{R}(x_\mathrm{R}, y)$ belongs to a foreground object. Thus a value significantly different from zero leads to the falsification of the hypothesis that the BDM is an accurate local description at a given scene point.

In reality, the result depends on the visual difference between background and foreground, between different foreground objects, and between points within foreground objects, as illustrated in Fig. 2. Suppose $\langle a_\mathrm{L}, a_\mathrm{R}, y\rangle$, $\langle b_\mathrm{L}, b_\mathrm{R}, y\rangle$, $\langle c_\mathrm{L}, c_\mathrm{R}, y\rangle$ and $\langle d_\mathrm{L}, d_\mathrm{R}, y\rangle$ are entries on the y^{th} scanline in the BDM. At object boundaries, segments $[(a_\mathrm{L}, y), (b_\mathrm{L}, y)]$ and $[(c_\mathrm{L}, y), (d_\mathrm{L}, y)]$ in the left image are mismatched against segments $[(a_\mathrm{R}, y), (b_\mathrm{R}, y)]$ and $[(c_\mathrm{R}, y), (d_\mathrm{R}, y)]$

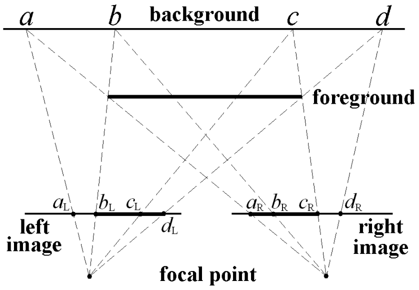


Figure 2: Background hypothesis falsification. Mismatch occurs at object boundaries and interiors.

in the right, respectively. In object interiors, segment $[(b_{\mathrm{L}},y),(c_{\mathrm{L}},y)]$ is mismatched against segment $[(b_{\mathrm{R}},y),(c_{\mathrm{R}},y)]$.

As most (non-camouflaged) real-world objects are texturally coherent, we find that foreground-background mismatches at object boundaries have higher intensity than those from object interior autodecorrelation, as visible in Fig. 9(b). Further, since boundary mismatches derive from the geometry of projection, they also have more regular shape. We now examine these boundary mismatches in detail.

2.2 Disparity Contours

At object boundaries, stereo mismatch arising from background hypothesis falsification forms contours, as illustrated in Fig. 3. Since disparity increases with proximity to the cameras, the width of the contour area in which background is mismatched against foreground depends on how poor the assumption of background was, in terms of depth error.

Consider, without loss of generality, the left view. According to Fig. 2 and Eq. (1), we have

$$\mathbf{x}_{\mathrm{L}}(a_{\mathrm{L}}) - \mathbf{x}_{\mathrm{R}}(a_{\mathrm{R}}) = \mathbf{d}_{\mathrm{B}}(a_{\mathrm{L}}). \tag{4}$$

Since b_{L} and a_{R} map to the same foreground point,

$$\mathbf{x}_{\mathrm{L}}(b_{\mathrm{L}}) - \mathbf{x}_{\mathrm{R}}(a_{\mathrm{R}}) = \mathbf{d}_{\mathrm{F}}(b_{\mathrm{L}}), \tag{5}$$

where $\mathbf{d}_{\mathrm{F}}(b_{\mathrm{L}})$ is the **foreground disparity** at b_{L}. Subtracting Eq. (4) from (5) and eliminating $\mathbf{x}_{\mathrm{R}}(a_{\mathrm{R}})$,

$$\mathbf{x}_{\mathrm{L}}(b_{\mathrm{L}}) - \mathbf{x}_{\mathrm{L}}(a_{\mathrm{L}}) = \mathbf{d}_{\mathrm{F}}(b_{\mathrm{L}}) - \mathbf{d}_{\mathrm{B}}(a_{\mathrm{L}}). \tag{6}$$

Similarly,

$$\mathbf{x}_{\mathrm{L}}(d_{\mathrm{L}}) - \mathbf{x}_{\mathrm{L}}(c_{\mathrm{L}}) = \mathbf{d}_{\mathrm{F}}(d_{\mathrm{L}}) - \mathbf{d}_{\mathrm{B}}(c_{\mathrm{L}}). \tag{7}$$

This means that the lengths of the segments $[(a_{\mathrm{L}},y),(b_{\mathrm{L}},y)]$ and $[(c_{\mathrm{L}},y),(d_{\mathrm{L}},y)]$ are exactly the differences between the foreground and background disparities at the object boundaries, and encode depth. Combining such segments vertically as in Fig. 3 will yield the depth-encoding contours of foreground objects, referred to as **disparity contours**. As the figure makes clear, the resulting contours lie at the left of object boundaries in the left image but at the right in the right image. There is thus no ambiguity in the boundary locations once the contours are extracted.

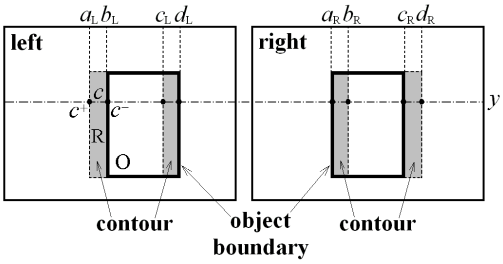


Figure 3: Disparity contours from background hypothesis falsification. Contour widths equal differential disparities between object and background.

2.3 Foreground Disparity Estimation

Foreground disparity can be estimated given the extracted disparity contours and the background disparities. Let c be a contour line segment in the left view of length $|c|$, and c^+ and c^- its left and right end points, as in Fig. 3. From Fig. 2 and Eq. (6), we have

$$\mathbf{d}(c) = |c| = \mathbf{x}_{\mathrm{L}}(c^-) - \mathbf{x}_{\mathrm{L}}(c^+) = \mathbf{d}_{\mathrm{F}}(c^-) - \mathbf{d}_{\mathrm{B}}(c^+), \tag{8}$$

where $\mathbf{d}(c)$ is the **differential disparity** between the background and foreground. We rewrite this equation, simplifying the notation without ambiguity, as:

$$\mathbf{d}_{\mathrm{F}}(c) = \mathbf{d}_{\mathrm{B}}(c) + \mathbf{d}(c), \tag{9}$$

which yields the foreground disparity at the boundary point. Let $\mathtt{R}$ be a contour region containing $|\mathtt{R}|$ such line segments. The average foreground disparity of $\mathtt{R}$ can be calculated by:

$$\bar{\mathbf{d}}_{\mathrm{F}}(\mathtt{R}) = |\mathtt{R}|^{-1} \sum_{c \in \mathtt{R}} \mathbf{d}_{\mathrm{F}}(c). \tag{10}$$

Similarly, the average disparity of an object O is:

$$\bar{\mathbf{d}}_{\mathrm{F}}(\mathrm{O}) = \frac{\sum_{\mathtt{R}\in\mathrm{O}} |\mathtt{R}| \bar{\mathbf{d}}_{\mathrm{F}}(\mathtt{R})}{\sum_{\mathtt{R}\in\mathrm{O}} |\mathtt{R}|}. \tag{11}$$

2.4 Foreground Hypothesis Verification (FHV)

Once disparity contours are extracted, we need to verify the potential objects delimited by the contours. Again, this can be done using disparity verification.

Let $\mathtt{R}_i$ and $\mathtt{R}_j$ be two vertically overlapping contour regions in the left view, i.e. $\pi_y(\mathtt{R}_i) \cap \pi_y(\mathtt{R}_j) \neq \emptyset$, as shown in Fig. 4. If there is a potential foreground object between $\mathtt{R}_i$ and $\mathtt{R}_j$, and assuming the depth

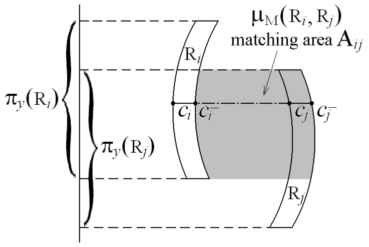


Figure 4: Foreground hypothesis verification by matching two disparity contour regions R_i and R_j, left view.

range of a foreground object is much smaller than the object-to-camera distance, its average disparity can be approximated, according to Eq. (11), by

$$\bar{\mathbf{d}}_{\mathrm{F}}(\mathrm{R}_i,\mathrm{R}_j)=\frac{|\mathrm{R}_i|\,\bar{\mathbf{d}}_{\mathrm{F}}(\mathrm{R}_i)+|\mathrm{R}_j|\,\bar{\mathbf{d}}_{\mathrm{F}}(\mathrm{R}_j)}{|\mathrm{R}_i|+|\mathrm{R}_j|}. \tag{12}$$

This foreground hypothesis can be verified by:

$$\mathrm{VDM}_{\mathrm{FHV}}(x_{\mathrm{L}},x_{\mathrm{R}},y)=|\mathrm{V}_{\mathrm{L}}(x_{\mathrm{L}},y)-\mathrm{V}_{\mathrm{R}}(x_{\mathrm{R}},y)|,$$
$$\text{where } (x_{\mathrm{L}},y)\in\mathrm{A}_{ij} \text{ and } x_{\mathrm{L}}-x_{\mathrm{R}}=\bar{\mathbf{d}}_{\mathrm{F}}(\mathrm{R}_i,\mathrm{R}_j). \tag{13}$$

For robustness, a **contour matching cost** is defined to normalize this result over the matching area A_{ij}:

$$\mu_{\mathrm{M}}(\mathrm{R}_i,\mathrm{R}_j)=\frac{\sum_{(x_{\mathrm{L}},y)\in\mathrm{A}_{ij}}\mathrm{VDM}_{\mathrm{FHV}}(x_{\mathrm{L}},x_{\mathrm{L}}-\bar{\mathbf{d}}_{\mathrm{F}}(\mathrm{R}_i,\mathrm{R}_j),y)}{\mathrm{area}\,(\mathrm{A}_{ij})}. \tag{14}$$

Then the foreground hypothesis between R_i and R_j is confirmed if $\mu_{\mathrm{M}}(\mathrm{R}_i,\mathrm{R}_j)$ is less than a threshold τ_{M}. Similarly, if an object is formed by grouping several contour regions, an **object cost** $\mu_{\mathrm{M}}(\mathrm{O})$ can be defined using the object's peripheral contours on the left and right sides to verify the object hypothesis.

2.5 Contour Grouping Direction

§2.4 provides an analysis of the ideal case of foreground verification. In reality, both background hypothesis falsification (BHF) and foreground hypothesis verification (FHV) depend on the amount of texture in the foreground and background. To understand the matter further, Fig. 5 plots the left view VDM results of BHF, FHV, and their subtraction along the y^{th} scanline of Fig. 3, assuming significant visual difference between foreground and background. The results are classified according to whether the background and foreground are textured or plain.

As can be observed, BHF distinguishes the foreground from the background only if the foreground is textured, and FHV does so only if the background is textured. However, the subtraction of the two, BHF−FHV, yields consistently higher values within the object area, between (b_{L},y) and (c_{L},y), than in the background area, left of (a_{L},y) and right of (d_{L},y), in three of the four cases. Even though the last case, where both the foreground and background are plain, would pose difficulties, it is statistically rare that the entire background and foreground areas remain textureless over time in a real environment. Therefore, by comparing BHF−FHV values in the left and right neighbourhoods of a contour region, we are able to determine in which direction, left or right, a contour region should be grouped with other contours.

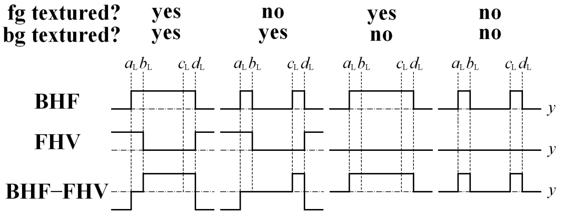

fg textured?
bg textured?
yes
yes
no
yes
yes
no
no
no
a_{L} b_{L} c_{L} d_{L}
BHF
FHV
BHF−FHV
y

Figure 5: The y^{th} scanline of view difference map (VDM), left view, where (a_{L},y), (b_{L},y), (c_{L},y), (d_{L},y) are the end points of disparity contour line segments, as in Fig. 3. The horizontal dashed lines indicate zero values and the solid lines indicate the VDM calculation results, simplified as positive, negative or zero. Top: $\mathrm{VDM}_{\mathrm{BHF}}$ from background hypothesis falsification; middle: $\mathrm{VDM}_{\mathrm{FHV}}$ from foreground hypothesis verification; bottom: $\mathrm{VDM}_{\mathrm{BHF}}-\mathrm{VDM}_{\mathrm{FHV}}$.

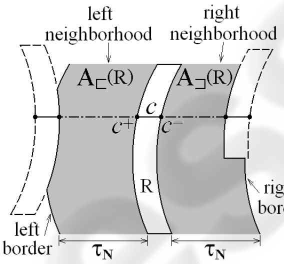


Figure 6: Left and right neighbourhoods of contour region R for computing contour grouping direction.

Let the left neighbourhood $\mathrm{A}_{\sqsubset}(\mathrm{R})$ of a contour region R be constrained by both the rightmost vertically overlapping contour region to the left of R and a distance threshold τ_{N}, whichever is closer, as illustrated in Fig. 6. Let $\mu_{\sqsubset}(\mathrm{R})$ denote the normalized subtraction result BHF−FHV in $\mathrm{A}_{\sqsubset}(\mathrm{R})$:

$$\mu_{\sqsubset}(\mathrm{R})=$$
$$\frac{\sum_{(x_{\mathrm{L}},y)\in\mathrm{A}_{\sqsubset}(\mathrm{R})}\mathrm{VDM}_{\mathrm{BHF}}(x_{\mathrm{L}},x'_{\mathrm{R}},y)-\mathrm{VDM}_{\mathrm{FHV}}(x_{\mathrm{L}},x''_{\mathrm{R}},y)}{\mathrm{area}\,(\mathrm{A}_{\sqsubset}(\mathrm{R}))},$$
$$\text{where } \langle x_{\mathrm{L}},x'_{\mathrm{R}},y\rangle\in\mathrm{BDM} \text{ and } x_{\mathrm{L}}-x''_{\mathrm{R}}=\bar{\mathbf{d}}_{\mathrm{F}}(\mathrm{R}). \tag{15}$$

Similarly, let $\mu_{\sqsupset}(\mathrm{R})$ denote the normalized BHF−FHV result in R's right neighbourhood $\mathrm{A}_{\sqsupset}(\mathrm{R})$. The **contour grouping direction** $\mu_{\mathrm{D}}(\mathrm{R})$ can then be calculated as:

$$\mu_{\mathrm{D}}(\mathrm{R})=\mu_{\sqsubset}(\mathrm{R})-\mu_{\sqsupset}(\mathrm{R}). \tag{16}$$

According to Fig. 5, R is on the left boundary of an object if $\mu_{\mathrm{D}}(\mathrm{R}) < 0$, on the right boundary if $\mu_{\mathrm{D}}(\mathrm{R}) > 0$, and within an object or background (or both the object and background are textureless) if $\mu_{\mathrm{D}}(\mathrm{R}) = 0$.

3 MULTI-OBJECT SEGMENTATION

Based on disparity contours, a multi-object segmentation system has been developed, as illustrated in Fig. 7. The system factors the segmentation problem into two stages: a well-understood offline stage and a novel online one.

Using the parameters of two calibrated cameras (Sun and Cooperstock, 2006), the offline stage constructs a background geometry model in the form of a background disparity map (BDM). This can be done by stereo matching (Kolmogorov and Zabih, 2002; Lin and Tomasi, 2004; Sun et al., 2003), structured light (Zhang et al., 2002), or ray tracing (Foley et al., 1997) from direct measurements of room geometry. The result is shown in Fig. 8.

The online stage compares new frames, captured, synchronized, undistorted and rectified, according to the pixel correspondence stored in the BDM to falsify the background hypothesis of the scene (BHF) and generate difference images, as shown in Fig. 9(b).

To extract disparity contours in a difference image D, a simple $[-1\ 1]$ edge operator is applied to generate an edge image, E. Clearly, positive edges are obtained on the left of a contour ridge in D and negative edges on the right. Therefore, positive and negative edge points are paired to form horizontal line segments, c, and their lengths, $|c| = \mathbf{d}(c)$, equal the differential disparities between foreground and background. Contour regions, R, are then formed by connecting these line segments vertically.

In order to remove noise caused by background model inaccuracies and foreground object internal texture, extracted contours go through an outlier removal step, including local line segment regularization and global region outlier removal. First, based on a Gaussian assumption, the horizontal line segments within each contour region whose lengths are outliers with respect to the region average are eliminated. Contour regions thus disconnected are reconnected by interpolation. Second, based on the observation that unwanted contour regions due to noise are usually small and of low intensity, and again assuming a Gaussian distribution for the two variables, the regions whose area and intensity are outliers with respect to the largest and brightest region are removed. Fig. 9(c) shows the final cleaned contours.

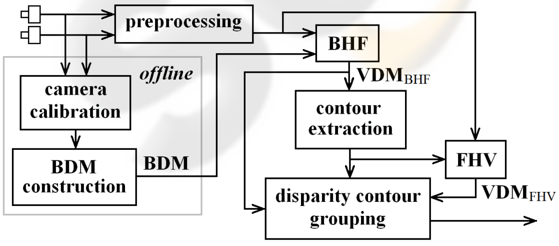


Figure 7: Multi-object segmentation system overview.

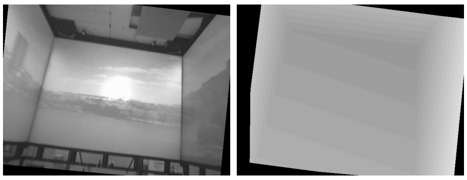

(a) (b)

Figure 8: (a) Left background image after undistortion and rectification. (b) Background disparity map (BDM) by ray tracing from 3D measurements.

Computing closed bounding object contours from bounding fragments relies on contour grouping, studied for many decades in perceptual organization (Elder and Goldberg, 2002; Elder et al., 2003). However, reliable contour grouping requires much computation and is unsuitable for real-time applications. We use a simple technique based on contour matching and disparity verification.

First, an initial grouping is performed to associate a contour region to its neighbours if they are close to each other and have similar average intensity and disparity. Then, based on contour grouping direction $\mu_{\mathrm{D}}(\mathrm{R})$, neighbouring contour regions with appropriate directions are selected for matching. If the matching cost $\mu_{\mathrm{M}}(\mathrm{R}_i, \mathrm{R}_j)$ is low, the regions are labelled to the same group. Finally, after all contour regions are grouped to objects, the object costs $\mu_{\mathrm{M}}(\mathrm{O})$ are evaluated and objects with high cost are eliminated as false. Fig. 9(d) demonstrates the result of contour grouping.

As explained in §2.2, disparity contours contain information about object boundary location in the input images. Therefore, objects can be segmented using the grouped contours, as shown in Fig. 9(e).

4 RESULTS AND ANALYSIS

The proposed segmentation system was tested in the augmented reality environment shown in Fig. 8(a). This space, which is representative of an important class of target environments, has a simple geometry,

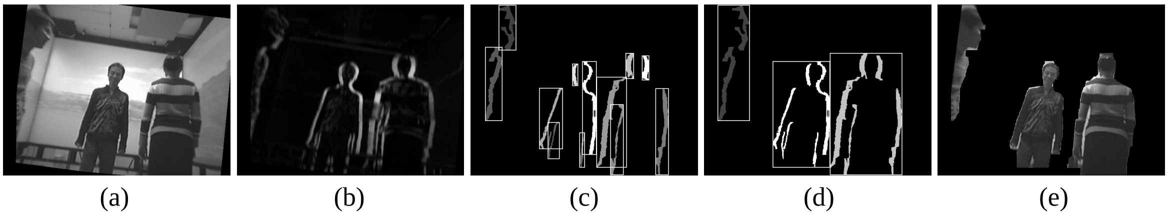


Figure 9: Processing sequence. (a) Original left view V_L after undistortion and rectification. (b) Corresponding left projection D_L of view difference map. High responses occur at object boundaries and within objects of non-uniform texture. (c) Extracted and cleaned disparity contours, augmented with contour region bounding boxes. Brightness represents average region intensity. (d) Contours grouped by matching and disparity verification. (e) Objects segmented by grouped contours.

easing BDM construction, and allows for dynamic retexturing of over 80% of the wall surface. Since, however, the method depends only on geometric stability, it is also applicable to more complex scenes.

A video with rapid changes in texture and illumination was projected onto the three screens surrounding the subjects. Two cameras on a small baseline were used and a grayscale image sequence of 1130 frames containing over 2000 foreground objects was captured. Sample images are shown in Fig. 10. As can be seen, the proposed method extracts multiple foreground objects despite complex changes in background texture.

In order to study the accuracy of object location, a quantitative analysis was conducted based on object bounding boxes, as shown in Table 1 and Fig. 11. The rate of ‘accurate’ object location, indicated by exact bounding boxes, with respect to the number of total objects reaches 60%, while the rate of ‘correct’ object location, including exact, noise enlarged, and partial object bounding boxes, totals 85%. Although partial occlusion, resulting in irregular contours, poses a challenge, the system still yields nearly 40% for ‘accurate’ object location and 55% for ‘correct’ location.

The thresholds on the matching cost $\tau_M = 20$ and the neighborhood distance $\tau_N = 50$, and an 11×1 block matching window for Eq. (3), were chosen empirically during algorithm development. No adjustment was required in testing. Further experiments (not detailed here) show that the method is quite robust to variation in these parameters.

Fig. 10 also compares our results to those of the graph cut algorithm (Kolmogorov and Zabih, 2002), one of the best stereo algorithms to date. Although graph cut produces acceptable scene disparity maps, its weakness on textureless regions, common in projected background, introduces many imperfections. Object segmentation based on this result would be challenging, requiring a large amount of post-processing.

Our unoptimized research implementation processes 640×480 monochrome image pairs at a rate of 3.8Hz (compared to graph cut’s 0.0023Hz) on a 1.8GHz 32 bit AMD processor. Our analysis suggests that an improved implementation can be structured to achieve performance comparable to only a few linear passes over the input data. Crucially, of course, the construction of the BDM is offline and does not contribute to the online processing time.

Although overall performance of the system is encouraging, some problems remain, due to both external errors and algorithm issues.

The foremost source of external error is imprecise environment calibration. Inaccuracies in the background model BDM introduce systematic background noise that confuses the segmenter and causes false objects. Using a special measurement device such as a laser pointer is expected to solve this problem. Other external errors such as camera synchronization error and video deinterlacing artifacts, which are further amplified by image undistortion and rectification, could be eliminated by employing progressive scan video cameras that take clock inputs.

Issues related to the algorithm itself include misleading contour grouping direction arising from texturelessness in both background and foreground, partial occlusion, the sensitivity of block matching to differences in camera response, viewing angle and specular lighting, and the dependence of boundary detection upon local intensity difference between background and foreground. However, based on the high success rate already achieved, exploiting the temporal coherence in an image sequence and adopting a higher-level tracker to propagate good segmentation results holds promise in all these areas.

Finally, the nature of the horizontally positioned stereo system results in a failure to detect horizontal or near horizontal object contours, such as at the top of the head and on the shoulders, and we have yet to investigate performance on highly textured foreground objects such as clothes with strong vertical patterns. However, adding vertical stereo into the framework and combining results on both axes can be expected to resolve both these concerns.

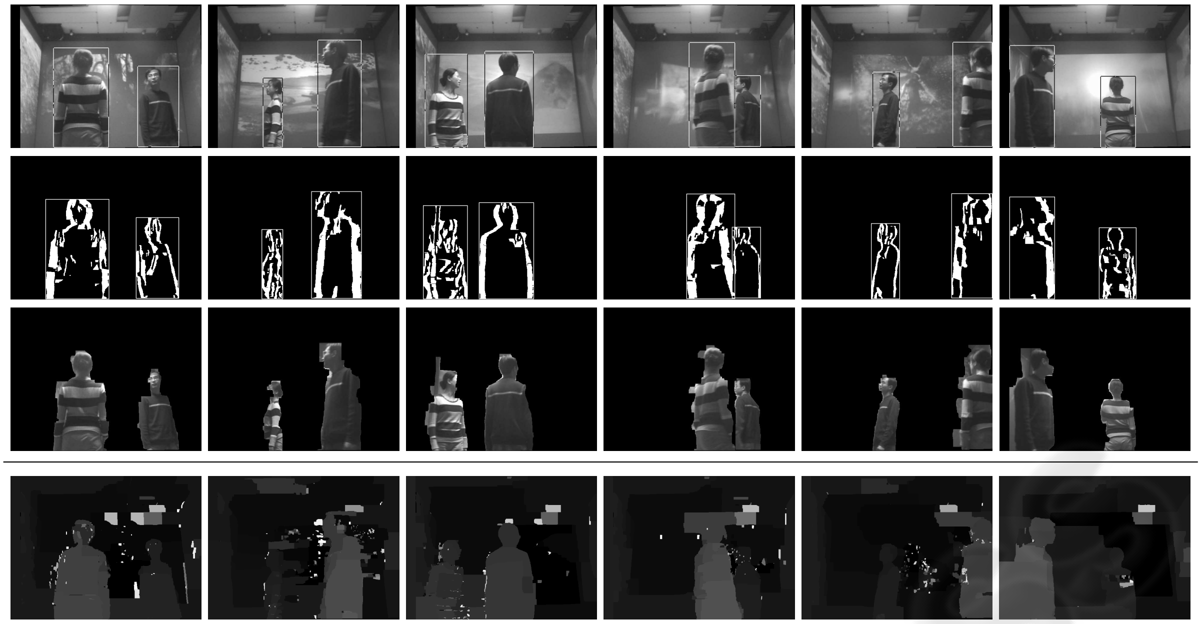
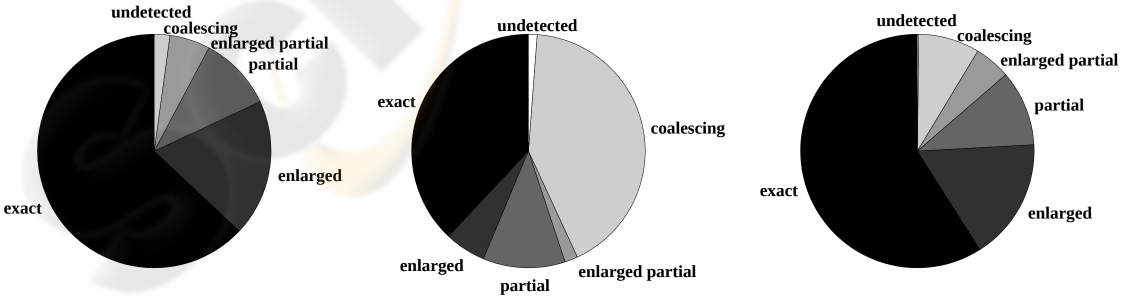


Figure 10: Samples of multi-object segmentation in the presence of fast lighting and texture changes. Top row: original images after undistortion and rectification, with bounding boxes indicating segmentation results. Second row: contour grouping results. Third row: objects segmented from the scene. Last row: comparison with scene disparity map by graph cut (Kolmogorov, 2003).

Table 1: Object location accuracy with respect to the number of total objects.

		no occlusion		partial occlusion		all objects	
total true objects		1767	100.00%	334	100.00%	2101	100.00%
accurate	exact bounding box	1113	62.99%	127	38.02%	1240	59.02%
inaccurate	enlarged bounding box	335	18.96%	19	5.69%	354	16.85%
	partial object	182	10.30%	38	11.38%	220	10.47%
incorrect	enlarged partial object	99	5.60%	6	1.80%	105	5.00%
	coalesced object	38	2.15%	140	41.92%	178	8.47%
	object undetected	0	0.00%	4	1.20%	4	0.19%
	false object					294	13.99%



(a) no occlusion (b) partial occlusion (c) all objects

Figure 11: Object location accuracy with respect to the number of total objects.

5 CONCLUSION

A new disparity contour grouping method to isolate and distinguish multiple foreground objects in a scene with fast illumination and texture change is presented. Without requiring full stereo reconstruction or tedious empirical parameter tuning, the method achieves near-real-time performance in software and generates not only the 2D image locations of objects but also boundaries and disparity information, providing a natural extension to 3D processing. As no assumption is made on the shapes and textures of objects and environment, the proposed approach suits generic object segmentation tasks.

ACKNOWLEDGEMENTS

The authors thank Jianfeng Yin for his help with room geometry measurement and video acquisition and Jeremy R. Cooperstock for providing essential research facilities.

REFERENCES

Ayer, S. and Sawhney, H. S. (1995). Layered representation of motion video using robust maximum-likelihood estimation of mixture models and MDL encoding. In *Int'l Conf. on Computer Vision*, pages 777–784.

Cucchiara, R., Grana, C., Piccardi, M., and Prati, A. (2003). Detecting moving objects, ghosts and shadows in video streams. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(10):1337–1342.

Elder, J. H. and Goldberg, R. M. (2002). Ecological statistics of Gestalt laws for the perceptual organization of contours. *Journal of Vision*, 2:324–353.

Elder, J. H., Krupnik, A., and Johnston, L. A. (2003). Contour grouping with prior models. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(25):1–14.

Foley, J. D., van Dam, A., Feiner, S. K., and Hughes, J. F. (1997). *Computer Graphics: Principles and Practice in C*. Addison-Wesley, 2 edition.

Fusiello, A., Trucco, E., and Verri, A. (2000). A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22.

Ivanov, Y., Bobick, A., and Liu, J. (2000). Fast lighting independent background subtraction. *Int'l Journal of Computer Vision*, 37(2):199–207.

Jepson, A. D. and Black, M. J. (1993). Mixture models for optical flow computation. In *Computer Vision and Pattern Recognition*, pages 760–761.

Kolmogorov, V. (2001-2003). Software. `http://www.adastral.ucl.ac.uk/~vladkolm/software.html`.

Kolmogorov, V., Criminisi, A., Blake, A., Cross, G., and Rother, C. (2005). Bi-layer segmentation of binocular stereo video. In *Computer Vision and Pattern Recognition*, pages 407–414.

Kolmogorov, V. and Zabih, R. (2002). Multi-camera scene reconstruction via graph cuts. In *European Conf. on Computer Vision*, pages 82–96.

Lin, M. H. and Tomasi, C. (2004). Surfaces with occlusions from layered stereo. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 26(8):1073–1078.

Narayanan, P. J., Rander, P. W., and Kanade, T. (1998). Constructing virtual worlds using dense stereo. In *Int'l Conf. on Computer Vision*, pages 3–10.

Oliver, N. M., Rosario, B., and Pentland, A. P. (2000). A Bayesian computer vision system for modeling human interactions. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(8):831–843.

Rittscher, J., Kato, J., Joga, S., and Blake, A. (2000). A probabilistic background model for tracking. In *European Conf. on Computer Vision*, pages 336–350.

Stauffer, C. and Grimson, W. (1999). Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition*, pages 246–252.

Sun, J., Zheng, N.-N., and Shum, H.-Y. (2003). Stereo matching using belief propagation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(7):787–800.

Sun, W. and Cooperstock, J. R. (2006). An empirical evaluation of factors influencing camera calibration accuracy using three publicly available techniques. *Machine Vision and Applications*, 17(1):51–67.

Torr, P. H., Szeliski, R., and Anandan, P. (2001). An integrated Bayesian approach to layer extraction from image sequences. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 23(3):297–303.

Toyama, K., Krumm, J., Brumitt, B., and Meyers, B. (1999). Wallflower: principles and practice of background maintenance. In *Int'l Conf. on Computer Vision*, pages 255–261.

Wang, J. Y. and Adelson, E. H. (1993). Layered representation for motion analysis. In *Computer Vision and Pattern Recognition*, pages 361–366.

Weiss, Y. and Adelson, E. H. (1996). A unified mixture framework for motion segmentation: incorporating spatial coherence and estimating the number of models. In *Computer Vision and Pattern Recognition*, pages 321–326.

Wren, C. R., Azarbayejani, A. J., Darrell, T. J., and Pentland, A. P. (1997). Pfinder: real-time tracking of the human body. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):780–785.

Zhang, L., Curless, B., and Seitz, S. M. (2002). Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *Int'l Symposium on 3D Data Processing Visualization and Transmission*, pages 24–36.