In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (Fort Collins, CO), June 1999.

Background estimation and removal based on range and color

G. Gordon, T. Darrell, M. Harville, J. Woodfill
Interval Research Corp.
1801C Page Mill Road
Palo Alto CA 94304
gaile@interval.com

Abstract

Background estimation and removal based on the joint use of range and color data produces superior results than can be achieved with either data source alone. This is increasingly relevant as inexpensive, real-time, passive range systems become more accessible through novel hardware and increased CPU processing speeds. Range is a powerful signal for segmentation which is largely independent of color, and hence not effected by the classic color segmentation problems of shadows and objects with color similar to the background. However, range alone is also not sufficient for the good segmentation: depth measurements are rarely available at all pixels in the scene, and foreground objects may be indistinguishable in depth when they are close to the background. Color segmentation is complementary in these cases. Surprisingly, little work has been done to date on joint range and color segmentation. We describe and demonstrate a background estimation method based on a multidimensional (range and color) clustering at each image pixel. Segmentation of the foreground in a given frame is performed via comparison with background statistics in range and normalized color. Important implementation issues such as treatment of shadows and low confidence measurements are discussed in detail.

1 Motivation

Separating dynamic objects, such as people, from a relatively static background scene is a very important preprocessing step in many computer vision applications. Accurate and efficient background removal is critical for interactive games[7], person detection and tracking[1, 4], and graphical special effects. One of the most common approaches to this problem is color or greyscale background subtraction. Typical problems with this technique include foreground objects with some of the same colors as the background (produce holes in the computed foreground), and shadows or other variable lighting conditions (cause inclusion of background elements in the computed foreground).

In this paper we present a passive method for background estimation and removal based on the joint use of range and color which produces superior results than can be achieved with either data source alone. This approach is now practical for general applications as inexpensive real-time passive range data is becoming more accessible through novel hardware[10] and increased CPU processing speeds. The joint use of color and range produces cleaner segmentation of the foreground scene in comparison to the commonly used color-based background subtraction or range-based segmentation.

Background subtraction based on color or intensity is a commonly used technique to quickly identify foreground elements. In current systems [3, 4, 11] performance is improved by using statistical models to represent the background (e.g single or multiple Gaussians at each pixel), as well as updating these models over time to account for slow changes. There are two classic problems with this approach. Clearly, if regions of the foreground contain similar colors as the background, they can be erroneously removed. Also, shadows cast on the background can be erroneously selected as foreground. This problem can be minimized by computing differences in a color space (hue, log color opponent, intensity normalized RGB[11]) which is less sensitive to intensity change. However, it is difficult to optimize a single match criterion such that it allows most shadowed pixels to match their normal background color and does not allow regions of the true foreground to match background pixels with similar hue. Figure 1 shows an example of color based segmentation failure.

Range has also been used for background removal[2, 5, 6]. The main issue in this approach is that depth computation via stereo, which relies on finding correspondences between two images, does not produce valid results in low contrast regions or in regions which can not be seen in both views. In our stereo implementation (described in section 2.1), these low confidence cases are detected and marked with a special value we will refer to as *invalid* . It

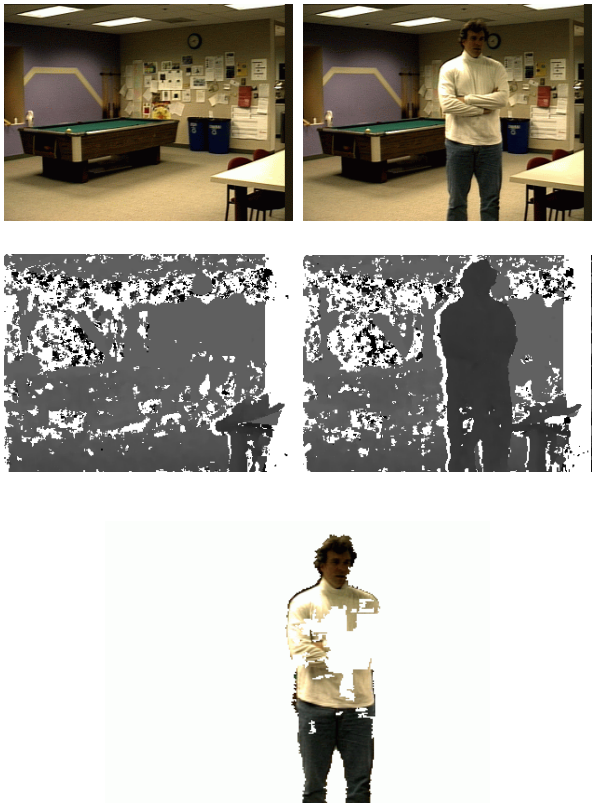

Figure 1. Color background subtraction has difficulty when portions of the foreground include the same colors as the background. Top left shows color background model, top right shows color image from scene. The bottom image shows segmentation results from comparison of these images. The range background model and image are also shown for reference, although they are not used in this segmentation.

Figure 2. Middle images show range background model and new scene image. Stereo computation can not produce valid range estimates in areas which have very low texture (e.g. saturated regions) or which are occluded in one view. Invalid range values are shown in white. Depth based segmentation, shown in bottom image, will fail in regions of the foreground which are undefined in depth. Top row shows color background model and scene image for reference, although they are not use in segmentation. Color of the foreground is overlayed on the segmentation results for easier interpretation.

is rare that all pixels in the scene will have valid range on which to base a segmentation decision. It is also difficult to use range data to segment foreground objects which are at approximately the same distance as the background. Figure 2 shows an example of range based segmentation failure.

We present a scheme which takes advantage of the strengths of each data source for background modeling and segmentation. Background estimation is based on a multi-dimensional (range and color) mixture of Gaussians which can be performed for sequences containing substantial foreground elements. Segmentation of the foreground is performed via background comparison in range and normalized color. For optimal performance, we find we must explicitly take into account low confidence values in range and color, as well as shadow conditions. The background estimation is described in section 2, followed by the segmentation method in section 3.

2 Background Estimation

In basic terms, we define the background as the stationary portion of a scene. Many applications simply require that there be introductory frames in the sequence which contain only background elements. If pure background frames are available, pixel-wise statistics in color and depth can be computed directly. The more difficult case is computing the background model in sequences which always contain foreground elements.

We model each pixel as an independent statistical process. We record the (R,G,B,Z) observations at each pixel over a sequence of frames in a multidimensional histogram. We then use a clustering method to fit the data with an approximation of a mixture of Gaussians. For ease of computation, we assume a covariance matrix of the form $\Sigma = \sigma^2 I$. At each pixel one of the clusters is selected as the background process. The others are considered to be caused by

foreground processes. In the general case where depth measurements at the pixel are largely valid, the background is simply represented by the mode which is farthest in range and covers at least $T\%$ of the data temporally. We use $T = 10$. In general, the required temporal coverage for good background estimation when depth is available can be much less than in a color only estimate because of the fact that background is inherently *behind* foreground. We need only insure that the deepest mode is a reliable process, and not due to noise.

However, if the pixel is undefined in range in a significant portion of the data (more than represented by the deepest mode) then we do not have sufficient data to model the background range and tag the range in the background as *invalid* (e.g. corresponding to a uniform distribution). We then cluster the data in color space and use the largest (most common) mode to define the background color.

As long as there is sufficient data representing the background at any given pixel over the sequence, the background can be estimated in the presence of foreground elements. In traditional color-based background estimation, which models the background color as the mode of the color histogram at each pixel, the background must be present at a given pixel in the majority of the frames for correct background estimation. A significant advantage of the use of color and depth space in the background estimation process is that, at pixels for which depth is usually valid, we can correctly estimate depth *and color* of the background when the background is represented in only a minority of the frames. For pixels which have significant *invalid* range, we fall back to the same majority requirement as color-only methods.

It is important to note the advantage of using a multi-dimensional representation. When estimating the background range or color independently, the background mode can be more easily contaminated with foreground statistics. Take for example, standard background range estimation[2] for a scene in which people are walking across a floor. Their shoes (foreground) come into close proximity with the floor (background) as they walk. The mode of data representing the floor depth will be biased to some extent by the shoe data. Similarly, in standard background color estimation, for a scene in which a person in a greenish-blue shirt (foreground) walks in front of a blue wall (background), the blue background color mode will be biased slightly toward green. However, assuming that the shoe is a significantly different color than the floor in the first case, and that the person walks at a significantly different depth than the wall in the second case, the combined range/color histogram modes for foreground and background will not overlap. This will result in more accurate estimates of background statistics in both cases.

2.1 Preprocessing of range data

Video from a pair of cameras is used to estimate the distance of the objects in the scene using the census stereo algorithm [12]. We have implemented the census algorithm on a single PCI card, multi-FPGA reconfigurable computing engine [10]. This stereo system is capable of computing 32 stereo disparities on 320 by 240 images at 57 frames per second, or approximately 140 million pixel-disparities per second. Using a commercial PCI frame-grabber, the system runs at 30 frames per second and 73 million pixel-disparities per second. These processing speeds compare quite favorably with other real-time stereo implementations such as [5].

The stereo range data typically includes some erroneous values which are not marked as *invalid*. These errors often take the form of isolated regions which are either much farther or much nearer than the surrounding region. Since the census algorithm uses a neighborhood based comparison when computing disparity between the two views, if an image region of uniform depth is small in comparison to the effective correlation window, disparities for the region are not likely to represent true distances in the scene. Therefore, before either background estimation or subsequent segmentation, we process the range to remove these artifacts using non-linear morphological smoothing[8, 9].

2.2 Background Estimation Results

In Figure 3, we show an example of the background computed from 60 frames sampled from a 780 frame sequence. The top row shows typical images from the sequence; there were no frames in the scene containing only background. The bottom row shows the background range and color representation in which all the foreground elements have been effectively removed.

These examples were computed with an off-line implementation of this background estimation algorithm. We are currently working on extensions which will allow dynamic background estimation based on the previous N frames (to allow for slow changes in the background), as well as an estimate of multiple background processes at each pixel, similar to [3], but using higher dimensional Gaussians.

3 Segmentation

Once we have an estimate of the background in terms of color and range, we can use this model to segment foreground from background in a subsequent image of the same scene. Ideally a pixel would be part of the foreground, $\mathcal{F}$, when its current value is far from the mode of the background model relative to the standard deviation.

$$\mathcal{F} \equiv |\mathbf{P_i} - \mathbf{P_m}| > k\sigma,$$

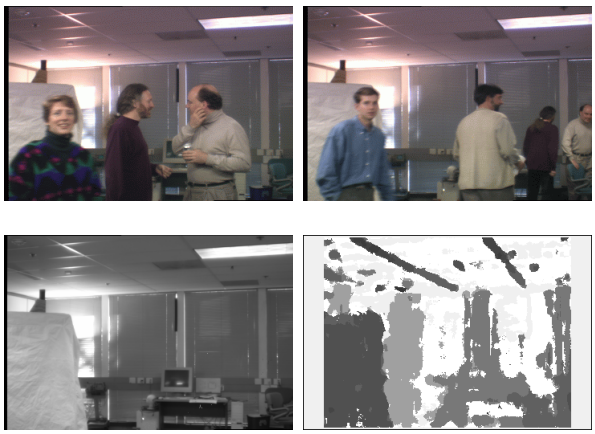

Figure 3. The top row shows sample images from a 780 frame sequence which contained no frames without people in the foreground. The bottom row shows the background model estimated from this sequence. These examples use an intensity and range model space.

where $\mathbf{P_i}$ is the pixel value at frame i (in color and range space), $\mathbf{P_m}$ is the mode of the background model at the same pixel, σ is the variance of the model at that pixel, and k is threshold parameter.

However, we must also take into account low confidence values, as well as the effect of shadows. The treatment of low confidence values is slightly different for range and color comparisons. At each pixel we will describe conservative foreground criteria, $\mathcal{F}_r$ and $\mathcal{F}_c$ for range and for color respectively based on the above general case. Then our final segmentation is a disjunction of the two criteria. The following sections describe the use of range, color, and their combination in more detail. Results of the combined segmentation are compared with using only range or color.

3.1 Use of Range

The presence of low confidence range values, which we have been referring to as *invalid*, in either the image or in the background model complicates our segmentation process. The most conservative approach would be to discount range in the segmentation decision unless range values in both frame i and the model, r_i and r_m respectively, are valid. We actually allow foreground decisions to be made when r_m is *invalid* but r_i is valid and smoothly connected to regions where foreground decisions have been made in the presence of valid background data:

$$\mathcal{F}_r \equiv \mathrm{Valid}(r_i) \wedge (\nabla r_i < G) \wedge \neg(\mathrm{Valid}(r_m) \wedge (|r_i - r_m| < k\sigma)$$

where ∇r_i is the local gradient of r_i. Gradient values above G represent discontinuities in range, so this value is set based on the expected smoothness of foreground objects.

As is shown by Figure 5, using the background model we can correctly classify the table (refer to original scene image in Figure 1) as background even though it is at same depth as the person. Note that Z-keying methods would fail in this case [5].

3.2 Use of Color

Shadows of foreground elements will cause appearance changes on the background. With out special treatment these appearance changes will be included in the foreground segmentation, which is usually not desirable. We attempt to minimize the impact of shadows in several ways. First, we use a luminance-normalized color space, $(\frac{R_i}{Y_i}, \frac{G_i}{Y_i}, \frac{B_i}{Y_i})$, which reduces the differences between a background object and itself under lighting changes induced by shadows or interreflections. We will refer to the distance between a pixel's value and the model in this color space as $\Delta\mathrm{color}$. This color representation becomes unstable or undefined when the luminance is close to zero, hence we define $\mathrm{YValid}(Y) \equiv Y > Y_{min}$. Our primary criterion for foreground segmentation is $\Delta\mathrm{color}$ which essentially corresponds to a hue difference in the context of valid luminance. We augment this comparison with a luminance ratio criterion and a final luminance comparison in the context of invalid model luminance.

$$\begin{aligned}\mathcal{F}_c \quad \equiv \quad & (\mathrm{YValid}(Y_m) \wedge \mathrm{YValid}(Y_i) \wedge (\Delta\mathrm{color} > c\sigma)) \vee \\ & (\mathrm{YValid}(Y_m) \wedge ((\frac{Y_i}{Y_m} < \mathrm{shad}) \vee (\frac{Y_i}{Y_m} > \mathrm{reflect})) \\ & \vee (\neg\mathrm{YValid}(Y_m) \wedge (Y_i > \alpha Y_{min})).\end{aligned}$$

The luminance ratio criterion is true for a pixel whose luminance differs sufficiently from the background that it is unlikely to be a shadow or interreflection. A shadowed background value is usually darker than the modeled background. Interreflections can lighten a background, but this effect is usually not as strong as the darkening due to shadows, hence we allow separate luminance ratio limits $shad$ and $reflect$. The last clause allows for a segmentation decision even when the model has very low luminance if the image luminance value is substantially higher than Y_{min}. We use $\alpha = 2$. This approach is similar to that used in [11].

Range-based adaptive thresholding

As we mention above, we minimize the impact of shadows by using a luminance-normalized color space. However there still remains a tradeoff in setting $c\sigma$ to be tolerant of remaining artifacts from strong shadows as well as maintaining integrity of the true foreground regions. We alleviate this tradeoff by using depth information to dynamically adjust our color matching criterion. We modify this simple scheme by increasing the color threshold wherever

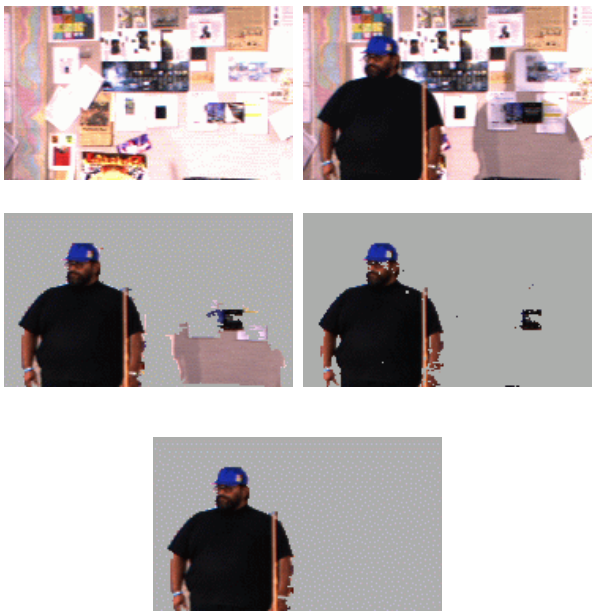

Figure 4. Top: Background image, person in foreground casting a strong shadow. Middle left: Basic color segmentation, shadow remains. Middle right: Effect on color segmentation when using the higher threshold for the entire image: skin tones close to background color are eroded. Bottom: large portions of shadow removed with adaptive (range-based) threshold.

the *depth* data indicates that a pixel belongs to the background. This has the effect of allowing us to be more lenient in our color matching within regions which appear to be at background depth, thereby allowing us to do a better job of ignoring shadows in these regions, while not compromising the restrictiveness of our color matching within regions in which depth is uncertain. (Note: Where depth indicates that a pixel is in the foreground, color matching is unimportant since the depth information alone is sufficient for correct segmentation.)

Figure 4 shows a case where a person casts a strong shadow on the wall. The middle left image shows the combined range and color-based segmentation when the color threshold is not adapted according to depth information. In this case, the shadow on the wall is sufficiently dark that it exceeds the color threshold setting, and causes the shadow to be labeled as foreground even though depth information indicates that it is background. If this color threshold is simply increased in order to remove the shadow (middle right image), valid parts of the foreground are eroded. The bottom image shows the combined range and color-based segmentation when the original color threshold is adaptively raised wherever the depth matches the background. The shadow is largely eliminated, while the remainder of the foreground is not impacted.

3.3 Combining Color and Range

We take a disjuction of the previous results to produce our final segmentation criteria, $\mathcal{F} \equiv \mathcal{F}_r \vee \mathcal{F}_c$. A pixel identified as foreground based on either depth or color is taken to be foreground in the combined segmentation.

This result will often contain small isolated foreground points caused by noise in color or range. There may also be some remaining small holes in the foreground. We fill the foreground holes using a morphological closing with a small structuring element. We can then take connected components over a certain minimum area as the final foreground segmentation result. The minimum area criteria can be set conservatively, to eliminate only noise related foreground elements, or it can be set at higher values based on the expected absolute size of "interesting" foreground elements, e.g. to select people and not pets.

3.4 Segmentation Results

The most compelling demonstration of this segmentation algorithm is to compare the segmentation results based on color or range alone with those achieved by the combined process. In particular, we use the examples presented in our introduction in Figures 1 and 2. Comparisons are presented in Figures 5 and 6 respectively.

We see that both cases produce more complete foreground segmentation. The holes present in range-based results are filled based on color comparison, and the holes present in color based results are filled based on range comparison. Using the joint segmentation approach, the only areas which would remain as problems are large regions with no valid range and colors similar to the background.

It is relevant to note that our use of range data does tend to produce a "halo" around foreground objects not present in the color only segmentation. Disparity maps produced by the census algorithm often include this halo effect in which pixels outside the perimeter of the foreground object are labeled as being at the depth of the foreground object. This error results from the fact that correlation-based stereo algorithms use windows much larger than a single pixel to determine correspondence, which works well in the case where the disparity for the entire window is constant. At depth discontinuities, the correlation window includes pixels with quite distinct disparities. Such depth discontinuities are often correlated with marked intensity change. Often this intensity change is the most significant feature in a correlation window. For a point just outside the perimeter of the foreground object, windows centered at the point in both views will share the significant intensity change and hence the point will be labeled as being at the depth of the foreground object. Although not presented here, we are also investigating the use of color discontinuities to correct for

Figure 5. Top row: range only segmentation, color only segmentation. Bottom: joint segmentation results.

Figure 6. Top row: range only segmentation, color only segmentation. Bottom: joint segmentation results.

the halo effect in range which slightly corrupts the silhouette boundaries in these results.

4 Conclusion

A simple, early method for background removal would be a useful step in many object recognition and tracking problems. We have demonstrated such a method based on the joint use of range and color data. This approach is quite compelling since fast, cheap (R,G,B,Z) sensors will be available commonly in the near future.

There are several advantages of this particular segmentation approach. The use of color and range together reduces the effect of classic segmentation problems in each data source when taken separately including: 1) points with similar color background and foreground, 2) shadows, 3) points with invalid data in background or foreground range, and 4) points with similar range background and foreground.

Background estimation in joint range and color space also presents several advantages. Higher dimensional histograms allow better separation of background and foreground statistics, resulting in a cleaner estimate at each point. The special interpretation of background as the farthest range event implies that at each point the background has to be visible in fewer frames for accurate background estimation. Background estimation in a scene which always contains some foreground elements is, in itself, a useful tool in site modeling and graphics.

References

[1] T. Darrell, G. Gordon. J. Woodfill, M. Harville, "Integrated person tracking using stereo, color, and pattern detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (Santa Barbara, CA), June 1998.

[2] C. Eveland, K. Konolige, R. Bolles, "Background Modeling for segmentation of video-rate stereo sequences," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (Santa Barbara, CA), June 1998.

[3] Grimson, Stauffer, Romano, and Lee, "Using adaptive tracking to classify and monitor activities in a site," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (Santa Barbara, CA), June 1998.

[4] Haritaoglu, Harwood, and Davis, "W4: Real time system for detecting and tracking people," *Proceedings of International Conference on Face and Gesture Recognition*, (Nara, Japan) April 1998.

[5] Kanade, Yoshida, Oda, Kano, and Tanaka, "A Video-Rate Stereo Machine and Its New Applications", *Computer Vision and Pattern Recognition Conference*, San Francisco, CA, 1996.

[6] Ivanov, Bobick, and Liu, "Fast Lighting Independent Background Subtraction". *Proceedings of IEEE Workshop on Visual Surveillance*, Bombay, India Jan 1998.

[7] Maes, P., Darrell, T., Blumberg, B., and Pentland, A.P., "The ALIVE System: Wireless, Full-Body, Interaction with Autonomous Agents". *ACM Multimedia Systems: Special Issue on on Multimedia and Multisensory Virtual Worlds*, Sprint 1996.

[8] J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, London, 1982.

[9] Luc Vincent, "Morphological Grayscale Reconstruction in Image Analysis: Applications and Efficient Algorithms", *IEEE Transactions on Image Processing*, 2:2, pp. 176–201, April, 1993.

[10] Woodfill, J., and Von Herzen, B., Real-Time Stereo Vision on the PARTS Reconfigurable Computer, *IEEE Symposium on Field-Programmable Custom Computing Machines*, Napa, April 1997.

[11] Wren, C.R., Azarbayejani, A., Darrell, T., Pentland, A.P., "Pfinder: Real-time Tracking of the Human Body", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19:7, July 1997.

[12] Zabih, R., and Woodfill, J., Non-parametric Local Transforms for Computing Visual Correspondence, *Proceedings of the third European Conference on Computer Vision*, Stockholm. May 1994.