Warping Background Subtraction

Teresa Ko Stefano Soatto Deborah Estrin
University of California, Los Angeles, CA 90095
{tko, soatto, destrin}@cs.ucla.edu

Abstract

We present a background model that differentiates between background motion and foreground objects. Unlike most models that represent the variability of pixel intensity at a particular location in the image, we model the underlying warping of pixel locations arising from background motion. The background is modeled as a set of warping layers, where at any given time, different layers may be visible due to the motion of an occluding layer. Foreground regions are thus defined as those that cannot be modeled by some composition of some warping of these background layers. We illustrate this concept by first reducing the possible warps to those where the pixels are restricted to displacements within a spatial neighborhood, and then learning the appropriate size of that spatial neighborhood. Then, we show how changes in intensity/color histograms of pixel neighborhoods can be used to discriminate foreground and background regions. We find that this approach compares favorably with the state of the art, while requiring less computation.

1. Introduction

Background subtraction is a common pre-processing step to many vision tasks such as object detection, localization, recognition, categorization, etc. In this context, a (foreground) "object" is defined as a compact region of space that is "different" from the background, and since the background is often modeled as a static map or distribution, any background *motion* triggers the detection of a novel *object*. However, in environmental monitoring scenarios, the background undergoes complex motions with self-occlusions that challenge these models even when the camera is not moving. Natural environments, such as the forest canopy, present a significant challenge because of the complex occlusion structure and motion of foliage, and the rapid illumination changes due to transitions between light and shadow (also an occlusion phenomenon). Clearly, representing or learning an accurate model of the background is not a viable proposition. Instead, we present a simple model that captures the phenomenology of background variations due to motion and occlusions for the purpose of detecting foreground objects within.

We define as "background" the portions of the scene that, over relatively long observation times, remain within the field of view, even though they may move and even disappear temporarily due to partial occlusions. Therefore, we model the background as a collection of *layers* (or "canonical images") that can move (undergo domain deformations, or "warpings"), and occlude each other to yield a generic background image. A foreground region, or "object," is thus another layer that cannot be obtained as a warping of a canonical image. Allowing permutations of layers effectively tightens the background distributions when there is significant background movement and intensity variation.

Unfortunately, finding the optimal unconstrained warping and layer combination that yield a sample image would be computationally infeasible. We therefore evaluate a number of possible techniques, each constraining the possible warping functions differently. Because background motion tends to be small (consider foliage moving in the wind), we limit the warping of image domains to a small spatial neighborhood, first heuristically, then by learning the distribution from the data. We implement two different functions to constrain this motion – a step function and a Gaussian window. When determination of the particular pixel warping is irrelevant, we propose a different approach using blocks of pixels. This "implicit" approach determines whether a patch in a sample image can be generated by warping a similar patch in the prototype background images, without representing such a warping explicitly. Using this latter technique, we obtain greater accuracy in background/foreground labeling with faster computation time.

We use bird monitoring in natural habitats as a motivating application to bring attention to a larger class of problems not previously addressed in the literature. A number of pertinent questions about the impact of climate change on our ecosystem are most readily answered by monitoring fine-scale interactions between animals and plants in their environment. Such fine scale measurements of species distribution, feeding habits, and timing of plant blooming

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
JUN 2010		**00-00-2010 to 00-00-2010**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Warping Background Subtraction	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of California, Los Angeles,Department of Computer Science,Los Angeles,CA,90095**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, June 2010. U.S. Government or Federal Rights License

14. ABSTRACT
We present a background model that differentiates between background motion and foreground objects. Unlike most models that represent the variability of pixel intensity at a particular location in the image, we model the underlying warping of pixel locations arising from background motion. The background is modeled as a set of warping layers, where at any given time, different layers may be visible due to the motion of an occluding layer. Foreground regions are thus defined as those that cannot be modeled by some composition of some warping of these background layers. We illustrate this concept by first reducing the possible warps to those where the pixels are restricted to displacements within a spatial neighborhood, and then learning the appropriate size of that spatial neighborhood. Then we show how changes in intensity/color histograms of pixel neighborhoods can be used to discriminate foreground and background regions. We find that this approach compares favorably with the state of the art, while requiring less computation.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**8**	

events require continuous monitoring, a task plagued by the very challenges described in this paper.

2. Prior Work

To detect novel objects in an image, many approaches model each pixel independently, $p(I_t(x)) \sim r_x \; \forall t$. W^4 [5] model r_x using the variance found in a set of background images with the maximum and minimum intensity value and the maximum difference between consecutive frames. Pfinder [24] learns the mean and the variance of pixel values at each location in the training set. If the mean and variance are all that is known about a distribution, the most reasonable choice of distribution based on maximal entropy is the Gaussian. The assumption then is $r_x = \mathcal{N}(\mu, \sigma^2)$, and a likelihood model is used to classify background and foreground at each pixel.

When this assumption does not adequately capture the distribution, a Mixture of Gaussians (MoG) can be used [20, 4] to further improve the accuracy of the estimate. A MoG model, where $r_x = \sum w_i \; \mathcal{N}(\mu_i, \sigma_i^2)$, is capable of handling a range of realistic scenarios, and thus is widely used [6, 21] to tackle the background subtraction problem. Elgammal *et al.* [3] show it is possible to achieve greater accuracy using a non-parametric model $r_x(i) = |I|^{-1} \sum_{t \in T} K(I_t(x) - i)$, where K is kernel function and i span the range of possible pixels value at the pixel x. Another contribution of this work is the incorporation of spatial constraints into the formulation of foreground classification. In the second phase of this approach, pixel values that can be explained by distributions of neighboring pixels are reclassified as background, allowing for greater resilience against dynamic backgrounds. Sheikh and Shah unify the temporal and spatial consistencies into a single model [18]. Similar models include [13, 16, 17]. Looking at the statistics at a single pixel shown in the central figure in Fig. 1, we see that the distribution of background pixels spans almost all grayscale intensities, and that the foreground distribution mostly overlaps. This indicates that there is a large overlap between background and foreground distributions, resulting in many false positives or misses.

A different approach, taken by Oliver *et al.* [15], looks at global statistics rather than local constraints. Similar to eigenfaces, a small number of "eigen-backgrounds" are created to capture the dominant variability of the background. The assumption is that the remaining variability in an image is due to foreground objects. The "eigen-background" approach works well for global changes in the background, such as variable illumination, but does not work well when the variability is local. If there are small changing regions in the background, as is the case in natural environments, the intensities of pixels A and B in Fig. 1 do not correlate, making "eigen-backgrounds" a poor model.

Yet another approach assumes that a background pixel is generated with a distribution that is based on its history, $I_t(x) \sim r_{1,\ldots,t-1}(x)$. The simplest of these models, frame differencing [7], thresholds the difference between two frames of a sequence, and large changes are considered foreground. To resolve ambiguity due to slowly moving objects, Kameda and Minoh [8] use a "double difference" that classifies foreground as a logical "add" of the pair-wise difference between three consecutive frames. A compromise between differencing neighboring frames and differencing against a known background image is to adapt the background over time by incrementally incorporating the current image into the background. Migliore *et al.* [12] integrate frame differencing and background modeling to improve overall performance.

Rather than implicitly modeling the background dynamics, many approaches have explicitly modeled the background as composed of dynamic textures [2]. Wallflower [22] uses a Wiener filter to predict the expected pixel value based on the set of past samples whose α's are learned. Monnett *et al.* [14] model the background as a dynamic texture [19], where the first few principal components of the variance of a set of background images (similar to [15]) comprise an autoregressive model in the same vein as [22, 9]. As shown in Fig. 1, pixels do not change in a predictable way over time, making dynamical models a poor fit for representing the background.

3. Approach

Our goal is to model the "usual" pixel values for background and detect the "unusual" pixels in image sequences captured from a fixed camera. We assume we start with a small number of training images, $\mathcal{T} = \{I_t(x) : t = 1, \ldots, T; x \in \Omega\}$, where $I : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^+; x \mapsto I_t(x)$, and the goal is to label all pixels in any image in the sequence, $I_t(x), \forall t > T$.

We assume that these images can be constructed from a canonical image $\hat{I}_0$ through some warping of the domain in $\hat{I}_0$. That is,

$$I_t(x) = \hat{I}_0(w_t(x)), \tag{1}$$

where $w_t \sim q$. The warping, w_t, is drawn from some displacement distribution q independently, so at any time t, any warping can be selected.

This model is valid only away from occlusions $\Theta \subset \Omega$ [1]. At occluded regions, a different scene is visible $\hat{I}_1$ that, in general, has no relation with $\hat{I}_0$. More generally, there can be an arbitrary number of occlusion layers, any of which can become visible at a given instant in time. Thus, rather than using a generative model derived from a single warped image, we can model the composition of several layers [23] each warped independently. A sample image is constructed by selecting the best warping from each canonical image, or

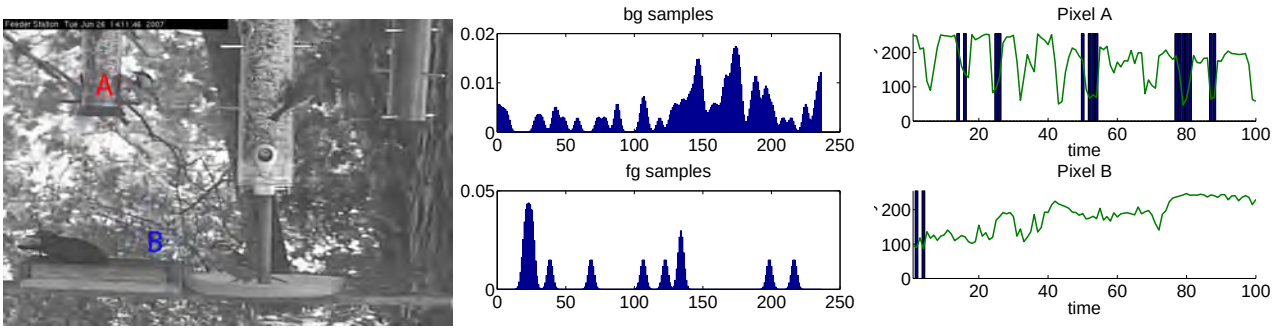

Figure 1. (Left) An image from a sample sequence. (Center) The distribution of pixel values at location A, separated into background (above) and foreground (below) pixels. Because of the range of background pixel intensities and the overlap of foreground and background distributions, modeling pixels individually would result in poor classification performance. (Right) The intensity variation of pixels A and B are not correlated over time, indicating that "eigen-background"-based methods do not capture these background changes.

layer, $\hat{I}_b$, such that,

$$\tilde{I}_t(x) = \sum_{b=1}^{B} \hat{I}_b(w_{t,b}(x))\chi_{b,t}(x), \tag{2}$$

where

$$\chi_{b,t}(x) = \begin{cases} 1 & \text{if } x \in \Omega \setminus \Theta_{b,t}, \\ 0 & \text{otherwise.} \end{cases}$$

But for an observed image I_t, we do not know if it contains foreground objects, how the background is warped (unknown $w_{t,b}(x)$) or where the occlusions occur (unknown $\chi_{b,t}(x)$). The pixel-wise discrepancy is thus:

$$D_t(x) \doteq \|I_t(x) - \sum_{b=1}^{B} \hat{I}_b(\tilde{w}_{t,b}(x))\tilde{\chi}_{b,t}(x)\|_2, \tag{3}$$

where $\tilde{w}_{t,b}(x)$ is the estimated warp and $\tilde{\chi}_{b_t}(x)$ is the estimated occlusion map. Depending on the application's tolerance for false positives versus missed detections, a threshold can be applied to the difference image for segmentation. The focus for the rest of this section is to model q adequately and in a computationally feasible way.

3.1. Modeling warp

In practice, not all warpings are plausible. Rather than allowing arbitrary warpings, we limit a pixel's possible warpings, Q, to its spatial neighborhood, where q is in the set Q, and

$$q(x) \sim Unif[x - \Delta x, x + \Delta x]. \tag{4}$$

We select a warping, w_t for I_t in a greedy fashion. For each pixel x, we find the best warping $w_{t,b}(x)$, restricted to pixels not previously warped from the canonical image $\hat{I}_b$ and to the square neighborhood specified by Δx. The "best" warping for each $\hat{I}_b$ is defined by the similarity of its appearance,

$$\tilde{w}_{t,b}(x) = \underset{w_{t,b}(x) \in Q \setminus \hat{Q}}{\arg\min} \|I_t(x) - \hat{I}_b(w_{t,b}(x))\|_2 \tag{5}$$

where $\hat{Q}$ is the set of q's that refer to previously matched x's. We then use the best warping from the set of canonical images as the unoccluded region,

$$\tilde{\chi}_{b,t}(x) = \begin{cases} 1 & \underset{b=1,..,B}{\arg\min} \|I_t(x) - \hat{I}_b(\tilde{w}_{t,b}(x))\|_2 = b, \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

This formulation can result in having no possible warp for a particular $I_t(x)$. That is, $Q = \hat{Q}$. In this case, since there are no pixels that can be matched, we assume the pixel is foreground, or an occlusion not modeled by the selected $\hat{I}_b$.

Using a uniform distribution around the pixel can result in poor matches when performing greedy matching. Since it is more likely that pixels are only warped slightly, we would like to bias our selected warps to those with minimal distance from the original location. To do this, we augment our minimization to:

$$\tilde{w}_{t,b}(x) = \underset{w_{t,b}(x) \in Q \setminus \hat{Q}}{\arg\min} \mathcal{G}_\sigma(x - w_t(x))\|I_t(x) - \hat{I}_b(w_{t,b}(x))\|_2 \tag{7}$$

where $\mathcal{G}_{\sigma^2}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-x^2/(2\sigma^2)}$.

Both approaches – using the uniform distribution and the Gaussian distribution – have parameters that can be learned from the data. In the uniform case, we can learn the appropriate Δx for each pixel x. For the Gaussian distribution, we can learn the appropriate σ^2.

3.2. Implicit warping

In reality, we are not interested in the precise warping of canonical images to sample image. Often, it is enough to know where the warping model fails, indicating foreground objects are present. Given the assumption that background motion is local, we can estimate how closely a patch of pixels in $\hat{I}_b$ matches those of I_t by measuring the distance of the *distribution of pixels* of each patch, instead of simply pixels.

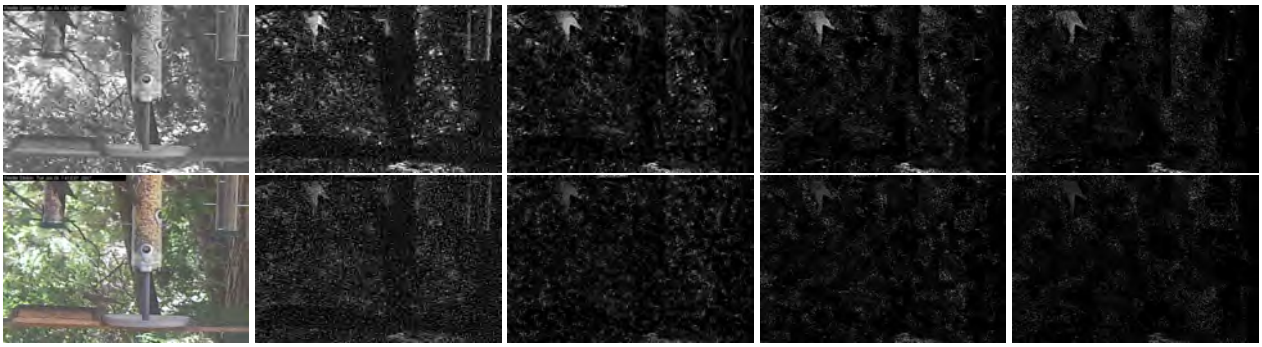

Figure 2. The top row shows the resulting difference per pixel using intensity as the feature, and the bottom row, using YUV as the feature. Left to right: 1) Raw image, I_t. 2-5) Difference image when $\Delta x = 1, 5, 9, 15$ respectively. Black pixels indicate a perfect match, white indicates no match, and the grays in between represent the "goodness" of the warp assigned.

We redefine the distance from the background, according to this function:

$$D_t(x) \doteq \min_{i=1,\ldots B} d(h_{x,I_t}, h_{x,\hat{I}_b}), \tag{8}$$

where $d(x, y) \doteq 1 - \sum_i \sqrt{x_i y_i}$, the inverse of the Bhattacharyya distance, and the histogram,

$$h_{x,I}(y) \doteq \frac{1}{w^2} \sum_{j \in J} \mathcal{G}_\epsilon(I(j) - y). \tag{9}$$

J is limited to the spatial neighborhood of x, so that $J = \{j : x - w/2 <= j < x + w/2\}$. The histogram, h is defined over the range of the image, $j \in [0\ 1]$. A Gaussian blur is used to smooth away the artificial edges induced by restricting subimages to non-overlapping blocks of the image.

4. Results

Detailed experiments are run on a 200 frame image sequence of birds at a feeder station from the data set released in [10]. We use 100 images for training and 100 images for testing. As this dataset has very few images of clean backgrounds, we use a ground-truth labeling of foreground/background pixels to exclude those foreground pixels from the training set of background images.

4.1. Basic Warp

We select five bird-less images from our training set as our canonical backgrounds, for the following experiments. We start by using a single canonical image, and vary the neighborhood in which we search for a warping match. Fig. 4 indicates that performance is hardly affected by different Δx values, whether we use the grayscale intensity as our feature, or color (in the YUV space). We compare these results to Elgammal's approach, and find that warping performs significantly worse. A closer look at the results shown in Fig. 2 indicates that the cause for such failure is the inaccuracy of the estimated warping. The bright white spots indicate pixels that could not be matched to the base image. As we increase Δx, shown consecutively from left to right, we see that more and more pixels are matched, but a significant number of pixels remain unmatched.

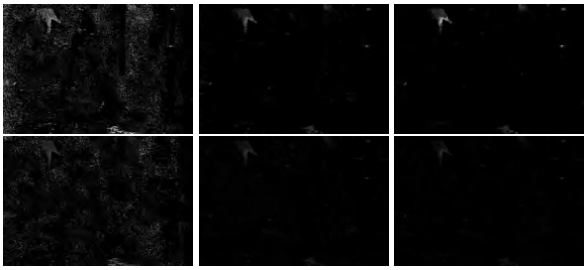

Figure 3. The top row shows the resulting difference images when using intensities as the feature, and the bottom row, using YUV as the feature as we increase the number of canonical images used. From left to right, we show $B = 1, 3, 5$, respectively. The addition of a single canonical image greatly reduces the number of unmatched pixels. There is little visible difference between $B = 3$ or 5, indicating that most occlusions are handled in the first 2 or 3 canonical images.

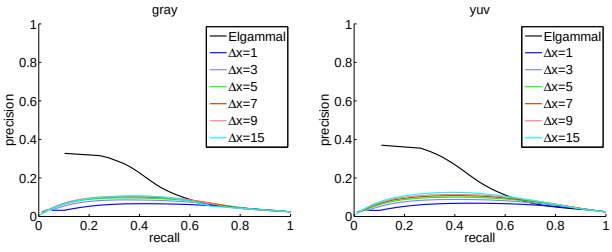

Figure 4. Restricting a pixel's warp to its spatial neighborhood results in rather poor performance, even as the radius of the neighborhood, Δx, is increased, regardless of the feature used (grayscale on the left, YUV on the right). The performance is significantly worse than Elgammal's approach, shown in black.

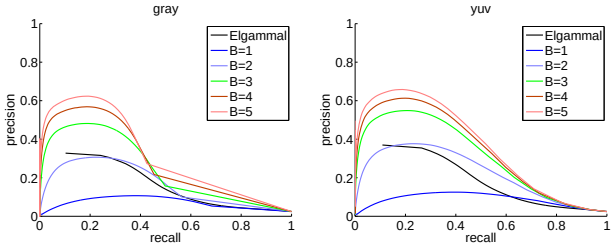

Figure 5. Occlusions are accounted for by using multiple base canonical images (where B is the number of images used). We see a significant performance improvement, as well as providing cleaner results than Elgammal's approach, for both gray (left) and YUV (right).

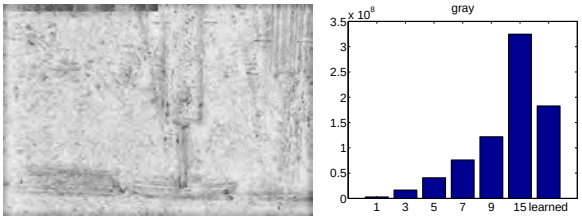

Figure 7. Learned step width, Δx, for each pixel for intensity (YUV yielded similar results). As expected, regions that are fairly stable, such as the feeder platform, have smaller step sizes (indicated by the darker color). Learning the appropriate Δx for each pixel maintains similar performance while reducing computation by half, as shown in the bar graphs.

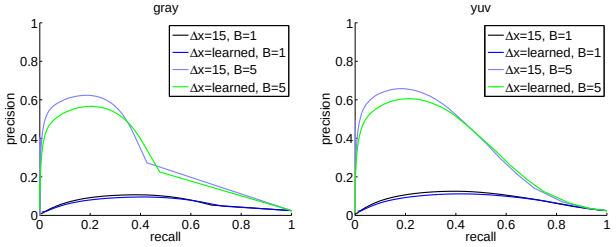

Figure 8. Learning Δx for each pixel results in very similar performance as compared to a fixed $\Delta x = 15$, regardless of the number of canonical images, B. $B = 1$ and $B = 5$ are shown here.

It is reasonable that many pixels are not matched, due to the large amount of background movement occurring in the sequence. Increasing the number of canonical images used, shown on Fig. 5, overcomes the problem of unmatched pixels, indicating that occlusion was indeed the limiting factor. As we increase the number of canonical images, B, we end up outperforming Elgammal's approach, in both the gray scale and YUV feature space. As indicated from Fig. 3, there are far fewer failed matches as we increase B.

Rather than defining a fixed Δx, we attempt to learn the appropriate Δx for each pixel to reduce computation. We start off by seeding the training algorithm with a manually

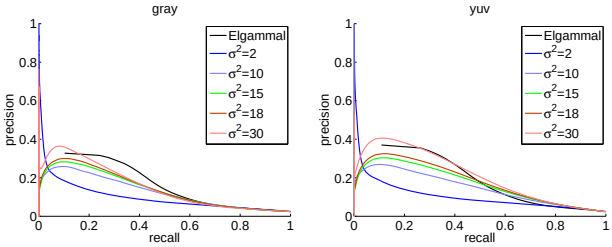

Figure 9. Using a Gaussian filter improves the performance as compared to Fig. 4. As expected, increasing the variance of the Gaussian window improves performance in both intensity feature space (left) and YUV (right).

selected canonical image. In this case, we use the first image in the sequence. Then, for each canonical image, we compute the best warp to each of the remaining training images. We discard warps to and from pixels that belong to foreground objects. We then select the next canonical image by choosing the training image that has the most unmatched pixels. Implicit in this assumption is that pixels that cannot be warped correspond to occluded pixels. Therefore, we select the image that reveals the most of the background that was occluded in the previously selected canonical images. We allow warping up to 15 pixels in any direction.

Fig. 7 shows the range estimated by this method. The left image shows the range learned where white pixels indicate the full ± 15 pixels and black pixels indicate a warp range of 0. This learned range results in half as many computations needed to estimate the warp than when Δx is fixed to 15, while maintaining a similar performance, as shown in Fig. 8. We show both precision recall curves with $B = 1$ and $B = 5$. The rate of improvement to the precision recall curve decreases as we go past 3 canonical images, indicating that most occluded backgrounds are modeled in the first 3 canonical images. This confirms our intuition that a few layers (leaves, feeder stations, sky) are sufficient to capture the phenomenology of the data.

4.2. Using a Gaussian window

We weight possible warping to regularize our matching scheme, making the resulting warp less sensitive to local minima. We test with several σ^2 and find that this greatly improves the performance as compared to the uniform warp shown in Fig. 4. Fig. 9 shows that, as we increase σ^2, we see a change in performance. With a large enough σ^2, we approach the accuracy of Elgammal's approach. Fig. 6 shows a smoother difference image, with few unmatched pixels, in both grayscale and YUV.

Adding base images (the same ones as used previously) results in improved performance. Similar to when a uniform warping was used, we achieve better performance as we increase B, as shown in Fig. 10.

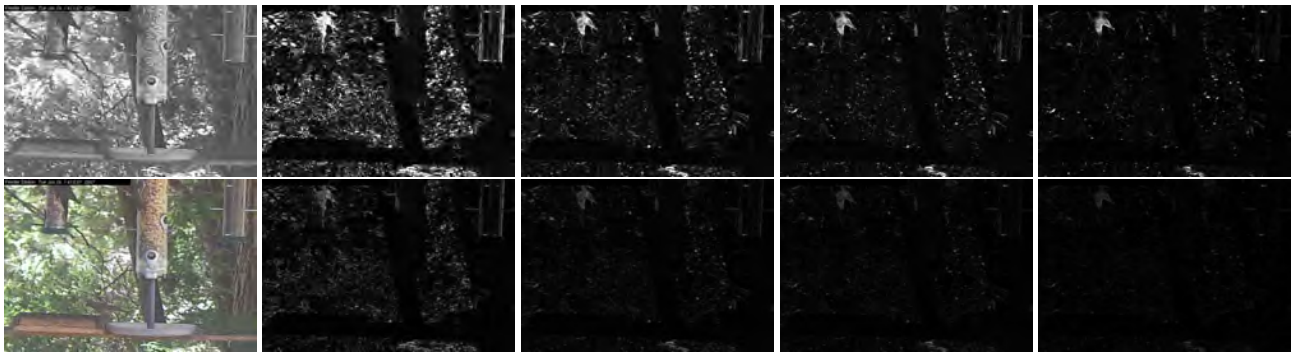

Figure 6. Resulting difference images using a Gaussian filter when selecting the best warping function, where the top row uses intensity as the feature, and the bottom row uses YUV. Left to right: 1) Raw image, I_t. 2-5) Difference image when $\sigma^2 = 2, 10, 15, 30$ respectively. As we increase σ^2, we see the moving foliage fade quickly into the background.

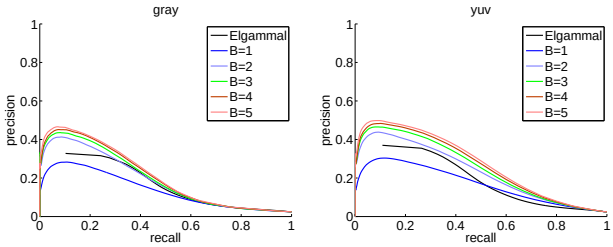

Figure 10. Occlusions are accounted for by using multiple base canonical images (where B is the number of images used), using the Gaussian window. This improves upon Elgammal's approach for both grayscale (left) and color (right) images.

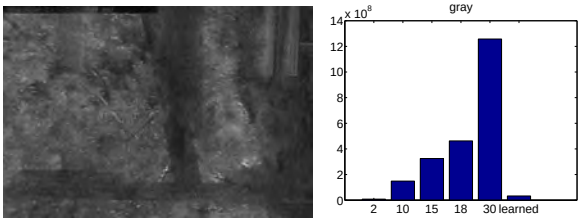

Figure 11. The σ^2 learned closely matches the underlying motion in the background. Darker pixels (small σ^2) appear where the background is fairly stationary, and lighter pixels (large σ^2) correspond to moving areas. The bar chart on the right shows the size of the search space. Using the learned σ^2 results in a search space that is orders of magnitude smaller.

We follow the same procedure used to learn the step radius Δx to learn the σ^2 for each pixel, and find that the results mirror Fig. 8. There is little performance loss but much greater computational efficiency, as shown in Fig. 11. The total search space (for all pixels), as illustrated in the bar chart, is reduced by an order of magnitude when the appropriate step size is learned.

Figure 12. Using the implicit warp model, the increasing patch width w reduces false positives, by enforcing a spatial warping across larger areas of the image.

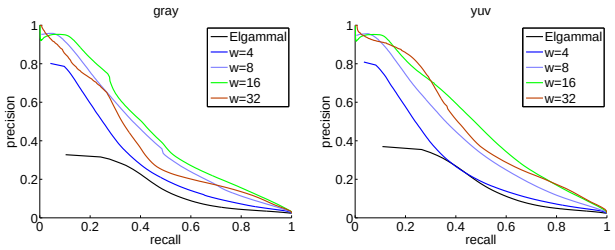

Figure 13. There is a tradeoff in selecting the appropriate block size, where w is the width of the block. As we increase w, we better handle background motion but we lose some precision because our granularity is now at the block level. Also, the foreground object contributes less to the distribution, resulting in less discrepancy between the background block and the block that is part of the foreground.

4.3. Implicit warp

We experiment with various patch widths, $w = \{4, 8, 16, 32\}$, and find that increasing w does not necessarily result in better performance as shown in Fig. 13. A closer inspection of the resulting difference images, Fig. 12, clarifies why this is so. When there is background movement, a larger block accounts for larger motion from background objects, such as the foliage of the tree. Yet, if the block is too large, foreground objects only contribute to a small part of the overall distribution, resulting in little change to $D_t(x)$.

Using multiple canonical images results in even better performance, as shown in Fig. 15. Though the effect is not

Figure 14. Adding canonical images accounts for the displacement of background objects. Note how the right feeder fades into the background noise as the number of bases increases.

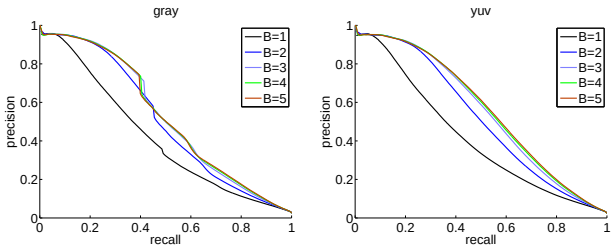

Figure 15. Adding canonical images, B, to the implicit warping model improves performance for both grayscale and color images.

as dramatic as in the other cases, we see that large displacements, such as the right feeder that swings back and forth, fades into the background as we increase B, as shown in Fig. 15.

4.4. Comparison to Prior Work

We compare our implicit warping model approach to several other approaches described in Section 2, and find that we mostly outperform the state of the art. Elgammal's approach [3] suffers because each background pixel exhibits a wide range of values, effectively making all possible values background. Sheikh's approach [18] is similar to our approach because it captures the local spatial neighborhood. But because it requires a locally consistent warping, it suffers from the same problem as Elgammal's approach, that each background pixel exhibits a wide range of values. Oliver's approach [15] does not model individual local motion, resulting in confused labeling where multiple motions occur.

This work builds on our previous work [10], but extends it by allowing multiple background layers while reducing computational complexity. Computationally, the proposed method is $O(B|I|)$, whereas our previous approach is $O(\Delta x^2|I|)$. Since patches are 30×30 pixels, computational savings can be one to two orders of magnitude. More methods, including dynamic texture subtraction, were shown to perform poorly for this data set in [10].

Looking at the average precision, our approach compares favorably on the image sequences released in [11]. The first 100 frames of each sequence are used for training and 20 images, labeled by [11]'s authors, are used for testing. The "Hall," "Lobby," and "Mall" image sequences contain people moving around indoor scenes that are fairly static. We see that Elgammal's and the explicit approach perform better than our previous approach and the implicit approach that has inherent smoothing. The remaining sequences consist of fairly large background motion from the object that is the name of the sequence (*e.g.*, the "Tree" image sequence has moving trees in the background). Both our explicit and implicit approaches outperform [3, 10] in these cases.

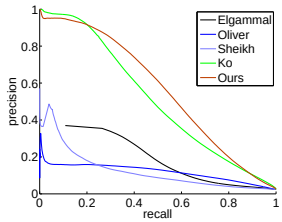

Figure 16. Our implicit warping model significantly outperforms approaches that model each pixel independently [3], spatial-appearance models [18], and linear combination approaches [15]. For the most part, it achieves better performance than [10], while requiring much less computation.

	[3]	[10]	Explicit	Implicit
Classic Image Sequences				
Hall	54.24%	22.13%	**67.11%**	40.64%
Lobby	11.66%	5.24%	**13.00%**	7.15%
Mall	**68.56%**	11.29%	62.24%	24.96%
Moving Background Image Sequences				
Trees	36.89%	64.92%	51.83%	**75.35%**
Curtain	86.89%	69.48%	87.94%	**94.09%**
Escalator	62.43%	19.04%	**64.55%**	62.49%
Fountain	47.33%	49.62%	57.83%	**71.21%**
Water	90.68%	53.07%	93.08%	**93.88%**

Table 1. While our approaches (both explicit and implicit) result in higher average precision than [3] and [10] in classic image sequences (people moving in indoor environments), the greatest effect is seen in sequences with moving backgrounds.

5. Conclusion

We propose a warping model to account for the displacement of pixels in the background image. We model the background as a set of canonical images to capture the different layers of background that appear or become occluded as background objects move. We find that the proposed approach better models the background in the case where there is significant motion, as demonstrated on image sequences of birds at a feeder station and more general, [11]. Furthermore, the implicit warping model performs better and requires less computation than the previous state of the art on this data set.

Acknowledgments. This material is based upon work supported by the CENS under the NSF Cooperative Agree-

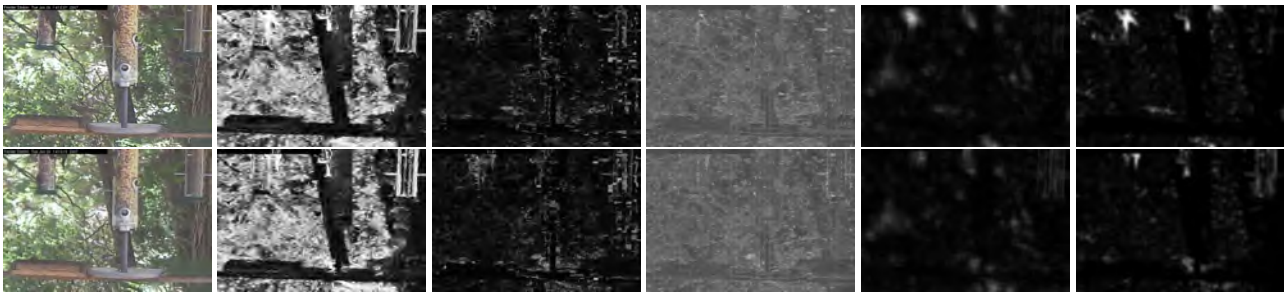

Figure 17. Sample difference images for the different approaches, from left to right: 1) Raw image. 2) Elgammal [3]. 3) Oliver [15]. 4) Sheikh [18] 5) Ko [10] 6) Our implicit warping model. We see that our approach better handles the background motion compared to (2-4) and is less blurred than (5). The relative difference between birds and the swinging feeder station is larger as well for our approach vs. [10].

ment CCR-012-0778 and #CNS-0614853, by ONR 67F-1080868/N00014-08-1-0414, ARO 56765-CI and AFOSR FA9550-09-1-0427. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NSF, ONR, ARO, AFOSR.

References

[1] A. Ayvaci, M. Raptis, and S. Soatto. Optical flow with occlusion detection as a convex optimization problem. In *Technical Report UCLA CSD 100013*, March 2010. 2

[2] G. Doretto, D. Cremers, P. Favaro, and S. Soatto. Dynamic texture segmentation. In *ICCV*, page 12361242, 2003. 2

[3] A. Elgammal, D. Harwood, and L. Davis. Non-parametric model for background subtraction. In *ECCV*, pages 751–767, 2000. 2, 7, 8

[4] N. Friedman and S. Russell. Image segmentation in video sequences: A probabilistic approach. In *13th Conference on Uncertainty in Aritficial Intelligence*, pages 175–181, 1997. 2

[5] I. Haritaoglu, D. Harwood, and L. S. Davis. W4: real-time surveillance of people and their activities. *IEEE Transactions on PAMI*, 22(8):809–830, 2000. 2

[6] M. Harville. A framework for high-level feedback to adaptive, per-pixel, mixture-of-gaussian background models. In *ECCV*, pages 543–560, 2002. 2

[7] R. Jain and H. Nagel. On the analysis of accumulative difference pictures from image sequences of real world scenes. *IEEE Transactions on PAMI*, 1(2):206–214, 1979. 2

[8] Y. Kameda and M. Minoh. A human motion estimation method using 3-successive video frames. In *International Conference on Virtual Systems and Multimedia*, pages 135–140, 1996. 2

[9] S. J. Kim, G. Doretto, J. Rittscher, P. Tu, N. Krahnstoever, and M. Pollefeys. A model change detection approach to dynamic scene modeling. *Advanced Video and Signal Based Surveillance, IEEE Conference on*, 0:490–495, 2009. 2

[10] T. Ko, S. Soatto, and D. Estrin. Background subtraction on distributions. In *ECCV*, volume 3, pages 276–289, 2008. 4, 7, 8

[11] L. Li, W. Huang, I. Y. H. Gu, and Q. Tian. Foreground object detection from videos containing complex background. In *MULTIMEDIA*, pages 2–10, 2003. 7

[12] D. Migliore, M. Matteucci, M. Naccari, and A. Bonarini. A revaluation of frame difference in fast and robust motion detection. In *VSSN*, pages 215–218, 2006. 2

[13] A. Mittal and M. Paragios. Motion-based background subtraction using adaptive kernel density estimation. In *CVPR*, pages 302–309, 2004. 2

[14] A. Monnet, A. Mittal, N. Paragios, and V. Ramesh. Background modeling and subtraction of dynamic scenes. In *ICCV*, pages 1305–1312, 2003. 2

[15] N. M. Oliver, B. Rosario, and A. P. Pentland. A bayesian computer vision system for modeling human interactions. *IEEE Transactions on PAMI*, 22:831–843, 2000. 2, 7, 8

[16] R. Pless, J. Larson, S. Siebers, and B. Westover. Evaluation of local models of dynamic backgrounds. In *CVPR*, pages II: 73–78, 2003. 2

[17] Y. Ren, C.-S. Chua, and Y.-K. Ho. Motion detection with nonstationary background. *Machine Vision and Applications*, 13:332–343, 2003. 2

[18] Y. Sheikh and M. Shah. Bayesian modeling of dynamic scenes for object detection. *IEEE Transactions on PAMI*, 27(11):1778–1792, November 2005. 2, 7, 8

[19] S. Soatto, G. Doretto, and Y. Wu. Dynamic textures. In *Proc. of the Intl. Conf. on Computer Vision*, pages 439–446, 2001. 2

[20] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. In *CVPR*, volume 2, pages 246–252, 1999. 2

[21] Y.-L. Tian, M. Lu, and A. Hampapur. Robust and efficient foreground analysis for real-time video surveillance. In *CVPR*, pages 1182–1187, 2005. 2

[22] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: Principles and practice of background maintenance. In *ICCV*, pages 255–261, 1999. 2

[23] J. Wang and E. Adelson. Representing moving images with layers. volume 3(5), pages 625–638, 1994. 2

[24] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on PAMI*, 19(7):780–785, 1997. 2