Improvements in Joint Domain-Range Modeling for Background Subtraction

Manjunath Narayana
narayana@cs.umass.edu

Allen Hanson
hanson@cs.umass.edu

Erik Learned-Miller
elm@cs.umass.edu

University of Massachusetts Amherst
Massachusetts, USA

Background subtraction, often a first step in segmenting moving objects in videos, is most commonly achieved by modeling the background color likelihoods at each pixel. Stauffer and Grimson [4] use a parametric Gaussian mixture model to estimate the likelihoods at each pixel. A non-parametric model was introduced by Elgammal *et al*. [1], where the likelihoods at each pixel are modeled using a kernel density estimate (KDE) by using the data samples from previous frames in history. These pixel-wise models do not allow for the observations at one location to influence the estimated distribution at a different but nearby location. By including each pixel's position information and modeling the likelihoods using a five-dimensional distribution in a joint domain-range representation, Sheikh and Shah [3] allow pixels in one location to influence the distributions in another location. They show that this sharing of spatial information leads to more accurate background subtraction. Their background model is a *single* distribution in the joint domain-range space. As we will see in this paper, their classification criterion, based on the ratio of likelihoods in this five-dimensional space, has an undesirable dependence on the size of the image. Like Sheikh and Shah, we model the foreground and background likelihoods with a KDE using pixel samples from previous video frames. However, we model the processes using a three-dimensional color distribution at each pixel. Our distributions are conditioned on spatial location, rather than being joint distributions over position and color. Our modeling avoids the dependence on the image size and yields better results.

Recent work on KDE based background modeling by Narayana *et al*. [2] has shown that adapting the kernel variance values for each pixel yields significantly better results than using a uniform kernel variance for all pixels. At each pixel location, the best kernel variance is selected from a set of candidate variances. Although we use a similar approach for adapting the kernel variance at each pixel, our background and foreground likelihood models are conceptually easier to interpret than their foreground and background *scores*. We show through both synthetic and real data examples that the adaptive kernel variance scheme is useful. With our probabilistic model, we can understand the effect of the adaptive kernel variance method of Narayana *et al*. more easily, as shown in Figures 1 and 2.

Another improvement we present over earlier approaches is the use of explicit spatial priors for the background and foreground processes. We use the foreground-background classification from the previous frame to estimate the prior probability for the processes. Our probabilistic formulation with likelihoods and a spatially dependendent prior for each process leads to a posterior distribution over the processes.

Figure 3 shows images that characterize the performance of the Sheikh and Shah model compared to ours and the effect of using adaptive kernel variance in both models. Benchmark comparisons on a standard data set show that our system's performance is comparable to the results of Narayana *et al*., which are the best reported results on our chosen benchmark. The advantage of our model over that of Narayana *et al*., is that our probabilistic model is more intuitive. The results from our model can be understood more clearly and the various constants and factors in the model can be interpreted more meaningfully.

[1] Ahmed M. Elgammal, David Harwood, and Larry S. Davis. Non-parametric model for background subtraction. In *ECCV*, 2000.

[2] Manjunath Narayana, Allen Hanson, and Erik Learned-Miller. Background modeling using adaptive pixelwise kernel variances in a hybrid feature space. In *CVPR*, 2012.

[3] Yaser Sheikh and Mubarak Shah. Bayesian modeling of dynamic scenes for object detection. *PAMI*, 2005.

[4] Chris Stauffer and W. Eric L. Grimson. Adaptive background mixture models for real-time tracking. In *CVPR*, 1999.

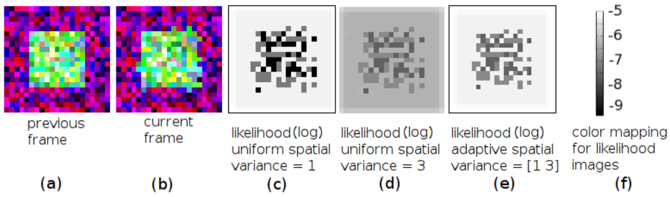

Figure 1: Consider a synthetic scene with no foreground objects, but to simulate spatial uncertainty, the colors in the central greenish part of the background in the previous frame (a) have been displaced at random by one or two pixel locations in the current frame (b). (c) Computing the background likelihoods at each location in the current frame with pixel samples from the previous frame using a small uniform variance results in low likelihoods for pixels that have moved. (d) Large uniform variance results in higher likelihoods of the moved pixels at the expense of lowering the likelihoods of stationary pixels. (e) Adaptive variance results in high likelihoods for both the moved and stationary pixels by applying a high spatial variance for pixels that have moved and a low spatial variance for pixels that have not moved.

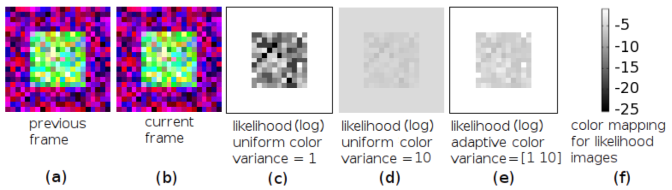

Figure 2: Uniformly sampled noise is added to the color values in the central part of image (a) to result in image (b). Color uncertainty in the central part of image (b) is best modeled by using adaptive kernel variances. (c) Small uniform variance results in low likelihoods for pixels that have changed color. (d) Large uniform variance results in higher likelihoods of the altered pixels at the expense of lowering the likelihoods of other pixels. (e) Adaptive variance results in high likelihoods for both kinds of pixels.

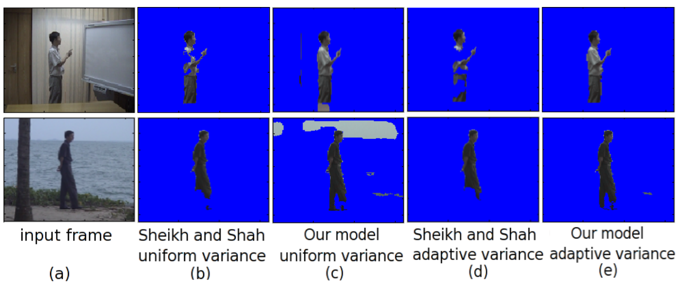

Figure 3: Comparing the effect of adaptive kernel variance for the Sheikh and Shah model versus our model. The Sheikh and Shah method has a bias towards background label (b), further exacerbated by the adaptive kernel selection (d). Our method tends to classify foreground objects well, but has more false positive foreground pixels (c). Adaptive kernel variance with our normalization yields the best results (e).The adaptive kernel for the background process, by selecting the best of the available kernel variances, in effect "tries hard" to classify each pixel as background. When a pixel is not well explained by the background model despite the selection procedure, it gets labeled as foreground.