Enriched Edge-Map Composite by Perceptual Fusion of Video Edge-maps.

Vishal Jain and Benjamin B. Kimia

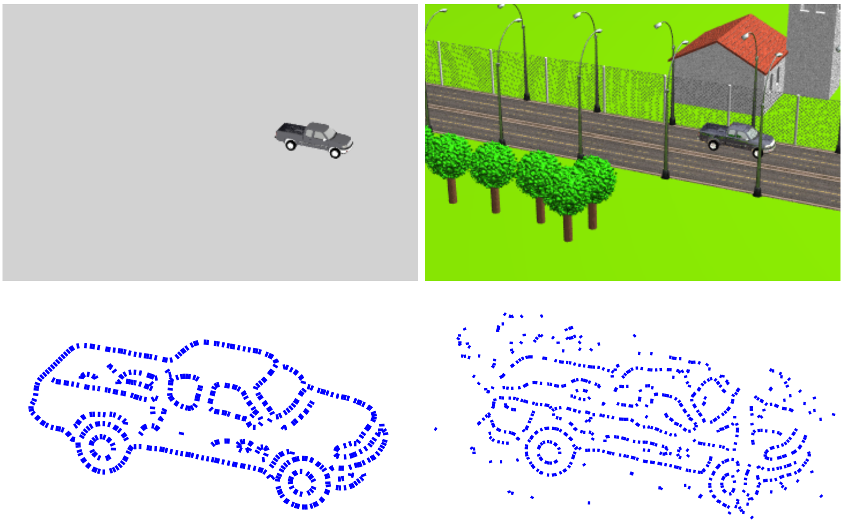
Division of Engineering,
Brown University,
Providence, RI, USA

Abstract. The detection of edges belonging to the foreground, even when using "background modeling" can be very challenging due to the blending of foreground with background, inter-reflections from surrounding objects, partial occlusions, *etc.*, leading to missing edges, spurious edges from background, highlights, and missing edges due to partial occlusion, *etc.* This renders the figure edge-maps after the background edges have been removed by background modeling unreliable and unusable. We propose an approach to this problem by integrating information across multiple adjacent frames (typically 5 or 7 frames). First, we align edge maps from neighboring frames to a central frame so as to obtain a compound edge-map. The alignment which "transports" the temporal information into a common reference is based on a view of an edge as a sample of an underlying curve. Second, we check the consistency of edges across frames by a notion of perceptual grouping based on geometric "edge consistency". This retains the edges which are consistent spatially as well as temporally and thereby removing some spurious edges and filling-in some gaps. Quantitative comparisons on synthetic video and qualitative comparisons on real video data shows that the resulting composite edge map is significantly better both on synthetic and on real data.

1 Introduction

Numerous computer vision applications such as surveillance, automated vehicles navigation, and robotics, among others, require segregated foreground objects. Foreground detection from a plain background with no illumination or inter-reflection effects, such as those in Figure 1(a) can be simple. However in a realistic scene, Figure 1(b), factors like illumination, multiple objects, occlusion, blending of foreground into background, *etc.* render the problem very difficult. For example, when true edge contrast falls below noise threshold, motion estimation can be erroneous, Figure 1. The use of background modeling of both edge location and orientation improves the results [4] but many of the problems remain. On the other hand, dynamic imagery like video provides additional and redundant information that we propose can lead to a more robust figure-ground segregation.

Our approach assumes that a background model is available and produces a figure edge map for each frame of the video. Our goal is to produce a composite edge map for each frame by integrating edge maps across several frames (typically 5 or 7 frames). Our main assumption is that significant edges (occluding contours, reflectance edges, shadows, *etc.*) typically persist over time, while most spurious edges arising from the coincidental alignment of intensities are not stable across frames. Interframe consistency can then be used in two complementary ways. First, the ability to deal with spurious edges through geomteric consistency across frames allows for the use of very low edge thresholds, which produce undesirable single frame edgemaps but which in composite form have fewer gaps and missing edges. Second, the disappearance of true edges in single frame edge maps are due to momentary occlusion (*e.g.* car behind a lamp-post), interreflections, coincidental alignment of intensity , *etc*, which are transitory phenomena. The use of multiple frames allows for the filling of such missing edges.



(a) (b)

Fig. 1. The bottom row shows edge maps of single frame shown in the top row. (a) When background modeling works perfectly, it is as if the object appears on a plain background, as the single frame from a synthetic video shows. However, typically in realistic imagery foreground detection suffers from numerous artifacts, *e.g.*, gaps spurious edges, highlight edges due to interreflections as shown in the same synthetic video frame but with a more realistic background (b).

Our Approach: We consider video acquired with a stationary camera[1] looking at a static scene with moving objects. We use the approach described in [4]

[1] A non-stationary camera could also be used but it would require a background registration of the video frames.

which uses an egde-based background model to detect the sub-pixel edge-maps of foreground objects where both edge locations and edge orientations employed resulting in improvements as compared to using edge locations only. Some of the detections are shown in the third row of Figure 2. Note the detections from individual frames are generally successful in differentiating figure from ground. However, there are also spurious edges and missing gaps due to the factors mentioned previously. The figure edge-maps have a lot of gaps, spurious edges, background edges and missing chunk of edges as illustrated in Figure 2. These problems are even more significant for lower-resolution imagery as the latter examples in the paper show.

We propose to first align the edge-maps from adjacent frames $\{I(t-n), I(t-n+1), ..., I(t-1), I(t+1), ..., I(t+n)\}$ (usually we use n is 2 or 3) onto the central frame $I(t)$ to form a "spatio-temporal"compound edgemap. This enables us to "transport" temporal information into the central frame. Second, we use geometric consistency of these spatio-temporal edges to distinguish structural vs spurious edges.

The paper is organized as follows: we first review some related work in Section 2. The geometric alignment of edge map is presented in Section 3 and the geometric consistency is presented in Section 4. The experimental results are described in 5.

2 Related Work

To the best of our knowledge, the idea of using edge maps both for the alignment/registration of frames and as a main feature for fusion across frames is novel. However, the idea of integrating edge information to fuse image or edge-maps from multiple sensors has been proposed earlier. Abidi and Delcroix [1] proposed an approach to fuse range and intensity edge-maps. They employ two ideas *(i) principle of token corroboration*: an edge in the final fused edge-map is retained if it is supported by either range and intensity edge-map and *(ii) principle of belief enhancement/withdrawl*: edge in the final edge map is weighted depending how similar is the edge content in both the edge-maps. Yocky [2] proposes fusion of multisensor images using wavelet transform. The idea is to fuse data which has compression along complementary datasets, *e.g.*, an image with high spatial resolution but low resolutioncolor information and another image with low spatial resolution but high color information. The authors enhance an image from a sensor using high frequency components from the other sensor image.

The work by Yang and Blum [3] proposes a method using multiple neighboring frames for fusion of multi-sensor images. The approach is to use a statistical model for image formation whose parameters as well as the final fusion image is unknown. An EM-based iterative algorithm is employed to solve for the parameters and the fused image iteratively. The temporal information or the neighboring frames add a constraint through consistency of parameters. The authors claim temporal information improves the fusion results.

3 Alignment of Edge-Maps

Our basis of method for registering two edge-maps is the work proposed by Chui and Rangarajan [5]. Aligning edge-maps requires estimating the correspondence between edge maps as well as the relative spatial transformation. The non-rigid matching using softassign algorithm in [5] first estimates the correspondence between two frames the correspondence and then the transformation between the corresponding point sets. The process is repeated to convergence. We extended/modified their existing algorithm for aligning edges in two ways: *(i)* pairwise point distance is replaced with point-curve distance and *(ii)* we employed the efficient Clough-Tocher interpolation scheme ($O(n \log n)$) instead of *CPU guzzler* thin plate spline ($O(n^3)$). First we summarize the approach in [5] and then discuss the modifications.

The correspondence is represented by a matrix M where columns represent edges in the first edge-map $\{e_i, i = 1, ..., K\}$ and rows represent edges in the second edge-map $\{\overline{e}_j, j = 1, ..., \overline{K}\}$. An additional column and an additional row are also added to represent outliers and missing edges, respectively. In binary form only one element in each row and each column can be one, indicating a one-to-one correspondence between the edges, if the one is not in the last column/row, or otherwise a spurious edge or missing edge, respectively. The matrix, however can accomodate a non-binary fuzzy representation if it is doubly stochastic [5]. Thus, an initial measurement of correspondence between the edges is considered. The transformation is a spatial map defined as a combination of affine transformation and thin plate spline function. Assuming correspondence between the two edge sets $\{e_i, i = 1, ..., L\}$ and $\{\overline{e}_j, j = 1, ..., \overline{L}\}$ is given, the transformation mapping an edge $e(x, y)$ to edge $\overline{e}(\overline{x}, \overline{y})$ is given by

$$\overline{e} = f_{A,\Omega}(e) = Ae + \sum_{i=1}^{L} U(|e - e_i|) w_i, \tag{1}$$

where A is an affine transformation matrix, and $U(r) = r^2 \log r^2$ and w_i are the weights of the thin plate spline (TPS) kernel Ω. The affine transformation brings corresponding edge elements into approximate registration while the thin plate spline transformation refines the registration.

The ideal transformation would be smooth while moving each edge to coincide with its corresponding edge. The free parameters of the transformation, A and Ω, are chosen to *(i)* minimize the distance between corresponding elements, *i.e.*, $|\overline{e}_i - f_{A,\Omega}(e)|$, and *(ii)* maintain a smooth map by minimizing the second derivatives of f leading to

$$E(A, \Omega) = \sum_{i=1}^{L} |\overline{e_i} - f_{A,\Omega}(e_i)|^2 + \lambda \iint \left[|\frac{\partial^2 f}{\partial x^2}| + 2|\frac{\partial^2 f}{\partial x \partial y}| + |\frac{\partial^2 f}{\partial y^2}| \right] dxdy, \tag{2}$$

where λ is a regularization parameter for the smoothness term.

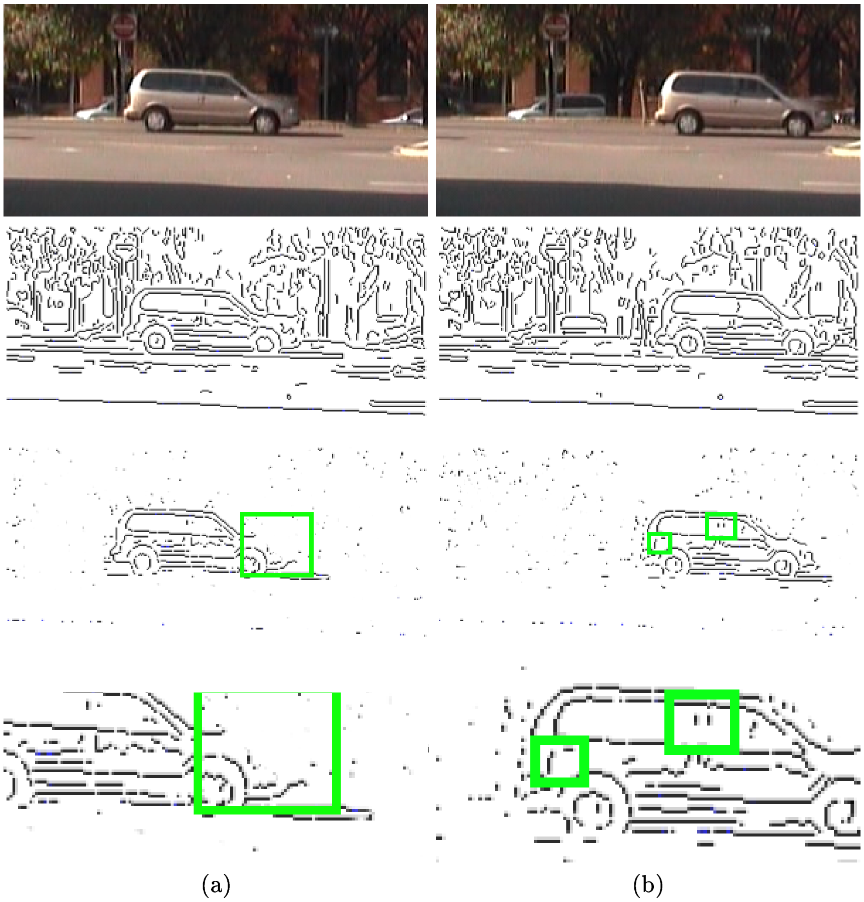


Fig. 2. The top row shows a couple of frames from a video, the second row shows the corresponding subpixel edgemaps, the third row shows the detected figure edges using [4]. Observe that in the third and fourth row (zoomed in), how gaps, *e.g.*, as boxed in (a) and spurious edges as boxed in (b) render local determination of figure from background from a single frame unreliable.

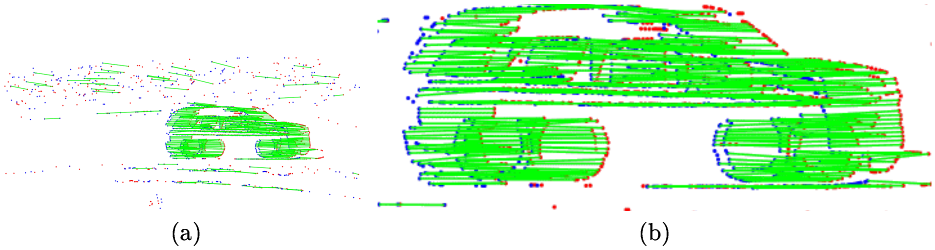


Fig. 3. (a) The correspondence between two edge maps is shown as green lines connecting corresponding edge points (red and blue) for two frames of an image sequence. (b) zoomed in (a). Observe that the majority of outliers are correctly deleted.

The two stages are combined in an iterative fashion, iterating between *(i)* finding the best transformation given the current correspondence M: Since M is not binary a weighted centroid $M[e_i]$ gives the effective edge position to correspond to $[\overline{e}_i]$; *(ii)* finding the best correspondence M given the current transformation of using $M_{ij} = e^{\frac{-|\overline{e}_i - f(e_i)|^2}{T}}$, where T is "temperature", M is then converted into a doubly stochastic matrix.

Initially, at a high temperature the elements of matrix M are assigned uniform values which implies all the pair-correspondences $\{e_i\} \times \{e_j\}$ are equally likely. As the temperature is lowered, M approaches a binary matrix which ensures one-to-one correspondence. Figure 3 illustrates the correspondence between two edge maps, where green lines connect points from one edge set (in red) to another edge set (in blue). The transformation allows for the edge maps from multiple frames to be superimposed on a single frame, Figure 8(d).

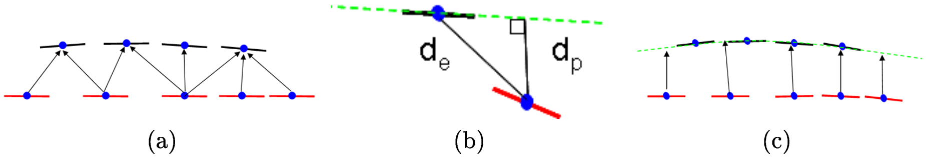


Fig. 4. This figure illustrates the advantage of using point to curve distance. (a) Point to point distance would be problematic as the sampling of the curve is different as shown in red and blue edges. (b) an estimate of point-curve distance and (c) point to curve distance allows the samples from other frames (red) to align with the underlying curve (shown in green).

Sampling variation issues: Consider a 2D curve C moving and deforming from one frame to another. This curve C is sampled differently in different frames giving rise to distinct spatial locations for edges in different frames. Figure 4(a) shows that using the point to point distance $d_p(i,j) = ||p_i - p_j||$ can lead to multiple correspondences and erroneous distnce estimates. We propose to estimate the point to curve distance instead. Since the curve from which the edge is sampled is not available, we use the best estimate, namely the line extension of the edge (when curvature information is available we use a circle). Thus, the distance of an edge e_i to the transported edge from another frame $\overline{e}_i$ is the distance between $\overline{e}_i$ and the line (or circle) extending e_i, Figure 4(b). This algorithm works well in general but (i) it sometimes produces erroneous correspondence due to variation in sampling across curves ans (ii) is computationally expensive. We now discuss our modification to address these problems.

Specifically the distance function between two edges then comprises three terms: (i) the perpendicular distance of an edge to the tangent of the other edge $d_p(i,j) = |(p_i - p_j) \times T_j|$, where T_j is the unit tangent of the edge $T_j = (\cos\theta_j, \sin\theta_j)$, and p_j is the edge position. (ii) The difference between orientation of edges $d_\theta = ||\theta_i - \theta_j||$ and (iii) the Euclidean distance between position of edges

$d_e = ||p_i - p_j||$ to define a local neighborhood over which the computation is meaningful. Then the similarity between two edges is represented by

$$s(i,j) = e^{-d_e^2/2\sigma_e^2} e^{-d_p^2/2\sigma_p^2} e^{-d_\theta^2/2\sigma_\theta^2}, \tag{3}$$

where σ_θ, σ_p and σ_e are the uncertainties associated with each of the distances and are typically assigned equal to 2.0 pixels, $\pi/6$ and 5.0 pixels respectively.
Efficient Implementation: Thin plate spline interpolation in [5] is used to model the elastic deformation between point-sets. Thin plate spline is computations are very expensive especially for the high number of edges retained after background modeling, typically of the order of thousands. It takes roughly 25 mins to register a pair of edge-maps(approx. 1000 edges). So this motivated to look for faster interpolation scheme which do not compromise the performance. Clough-Tocher implementation [6] of piecewise cubic patch is used to get the required speed up without giving up performance. The order of the algorithm is $O(n \log n)$ as compared to $O(n^3)$ of thin plate splines.

4 Geometric consistency of Spatio-temporal compound edge-map

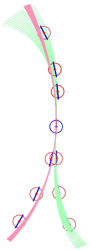


Fig. 5. Multiple groupings or curvelets shown in pink and green for an edge(blue circle). Each edge can have multiple groupings.

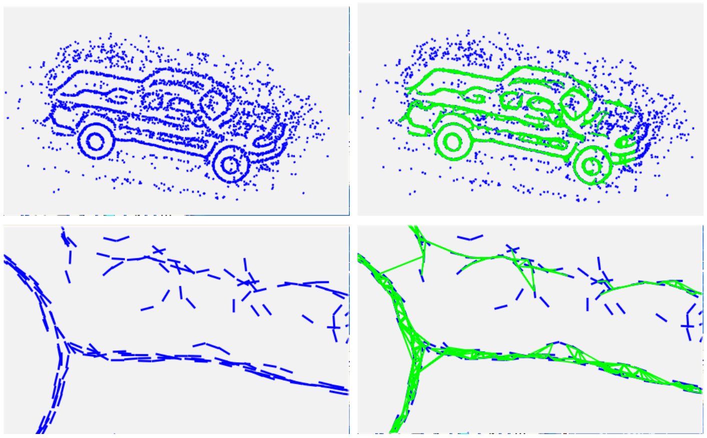


Fig. 6. This figure illustrates the perceptual grouping of curvelets on compound edge-map to retain the structural edges. The top row shows compound edge map (left) and the curvelets computed for it (right). The bottom row shows the zoomed and cropped image from the top row. Note that edges which do not support the curve model do not form any curvelets and hence are discarded.

By aligning and superimposing multiple edge-maps to a central frame we have transported the temporal information onto a single composite frame. We now discuss the issue of integration of edge information across frames in the composite edge-map. The gaps in the central frame would likely be completed by the edges from other frames and spurious edges would likely not receive support from edges in the other frames. In order to retain structural edges we need to check the geometric consistency of superimposed edges by perceptual grouping, as proposed in [7]. In this work authors propose the construction of "curvelets" which are local discrete combinations of edges which satisfy a local curve model, the Euler Spiral (constant rate of change in curvature), Figure 5. This is done by constructing discrete combinations from edges in a small (5×5) or (7×7) neighborhood through which an Euler Spiral passes.

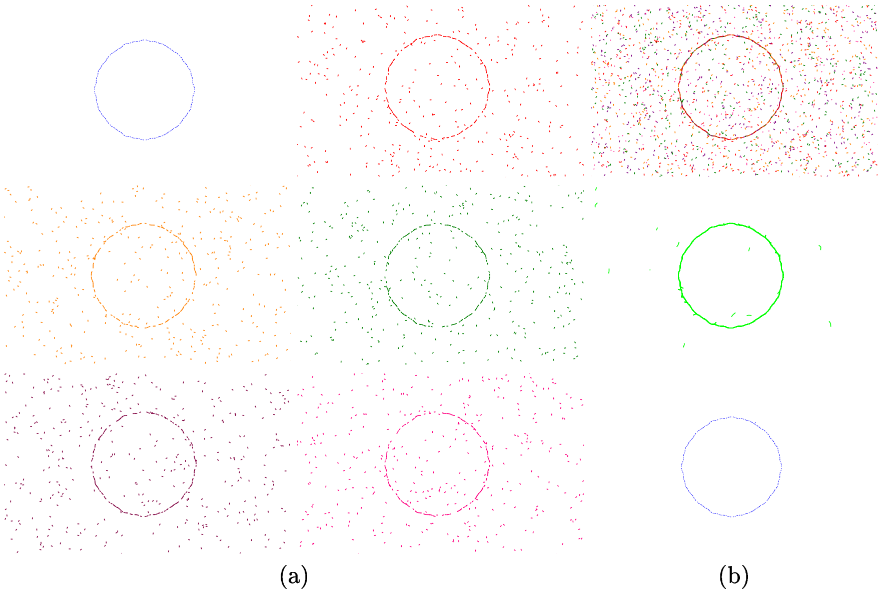


Fig. 7. This figure demonstrates the approach to recover structural edges using geometric consistency via curvelets. (a) Top left is the original edge-map, rest of the five edge-maps in different colors are obtained by adding noise and removing edges randomly in the original edge-map. (b) Top image is when all the noised edge-maps are superimposed, middle image shows curvelets for the superimposed edge-map and last one is the edge-map retained from the curvelets map.

More concretely, the algorithm considers each edge in turn as an anchor edge. Next construct pairs from each edge in a 5×5 neighborhood and the anchor edge. Each edge pair defines a family of Euler spirals. Those Euler Spirals in this family that pass through a third edge are retained and form a curvelet hypothesis. Additional information from a fourth edge further strengths the hypothesis. Typically, curvelets where the Euler Spiral passes through 6 or 7 edges are kept as viable. Figure 6 illustrates how this idea is applied to the fusion of edges in a video sequence. The superimposed compound edge map top left is fitted with curvelets shown in top right. Observe how there are no curvelets (green) formation for the spurious edges. Last row shows the zoomed and cropped images from the top row. This clearly shows curvelets are formed where there is lot of edges supporting or lying on some curve where as the edges which are scattered randomly do not form curvelets which we discard. We demonstrate this idea further on a simple data like a circle. The ideal image top left is viewed in five different frames with different edges missing and different spurious elements. The result of first aligning these frames and then find curvelets in the composite edge map is shown in the bottom right of Figure 7.

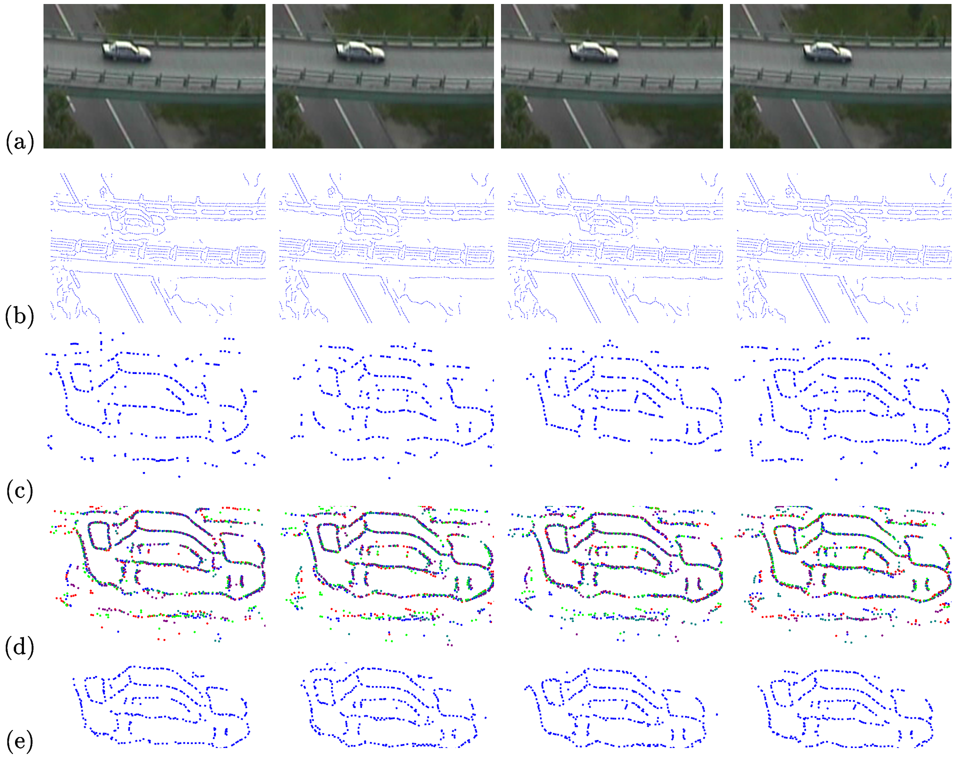


Fig. 8. Results of our approach on a video sequence. (a) Video sequence, (b) subpixel edge-map of (a), (c) Foreground edge-map, (d) superimposed edge-maps of 5 frames, (e) consistent edges retained. Note the difference between (c) and (e) figure edge-maps.

5 Experimental Results

We demonstrate the results of our approach on several synthetic and real videos. First, in Figure 8, we visually compare the foreground edge-maps from individual frames to the foreground edges obtained using our approach. Note how gaps in Figure 8(c) are closed using our approach, Figure 8(e), and how numerous spurious edges are discarded. The foreground edge-maps obtained using our approach look more complete than the original foreground edge maps. In order to highlight the strength of our approach, we also considered a video where the object undergoes partial occlusion and the foreground edge-map is incomplete and incorrect as it contains edges of the occluding object, as shown in Figure 9(a,c). Note how our approach not only fills in those gaps because of occlusion but also throw away the occluder's edges. It is clear that qualitatively our approach provides *more reliable and more complete* foreground edge-maps.

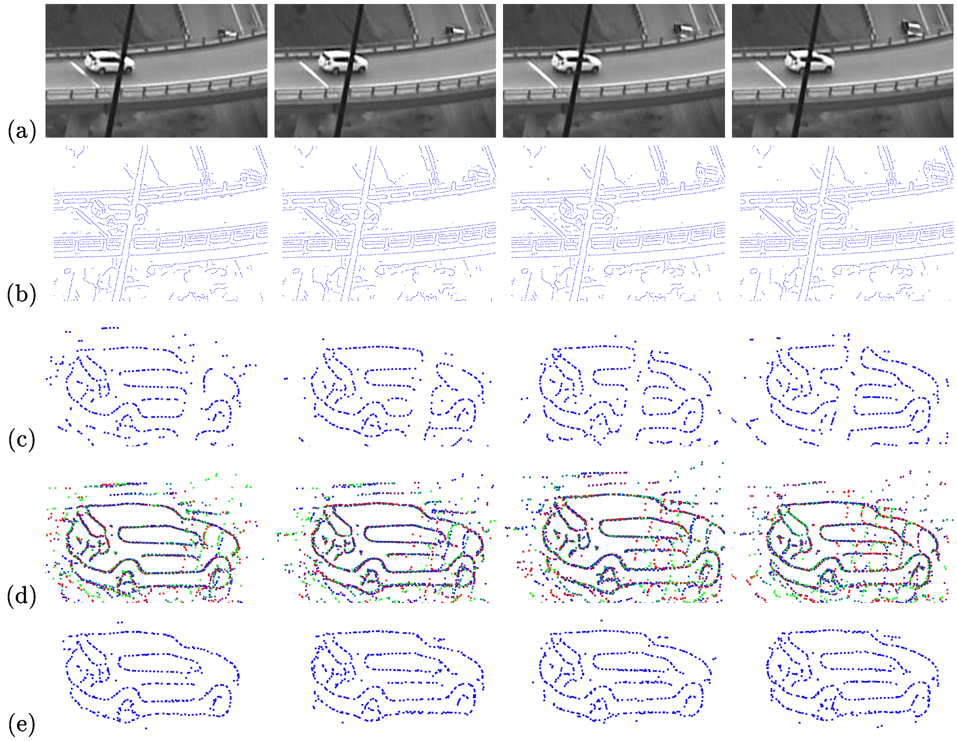


Fig. 9. Results of our approach on a video sequence under occlusion. (a) Video sequence, (b) subpixel edge-map of (a), (c) Foreground edge-map, (d) superimposed edge-maps of 5 frames, (e) consistent edges retained. Note the difference between (c) and (e) figure edge-maps. The occluded part is filled in.

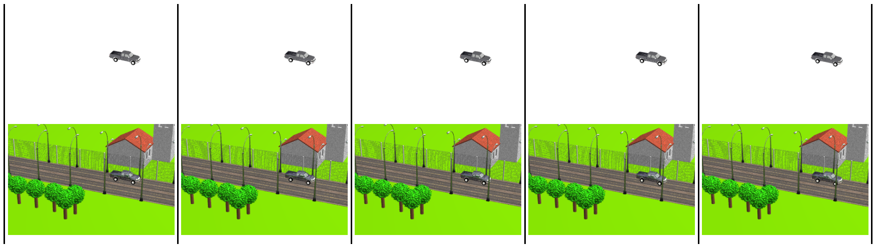


Fig. 10. This figure shows our synthetic video rendered using [8]. Top row shows a vehicle moving against a simple background and the bottom row shows the scene rendered with a more complex background with multiple objects, fences, posts, *etc.* to make it more realistic and model some of the factors like blending of object into background, inter-reflections, occlusion, *etc* which are responsible for degradation of the figure edge-maps. Figure 1 shows that our examples achieves that purpose.

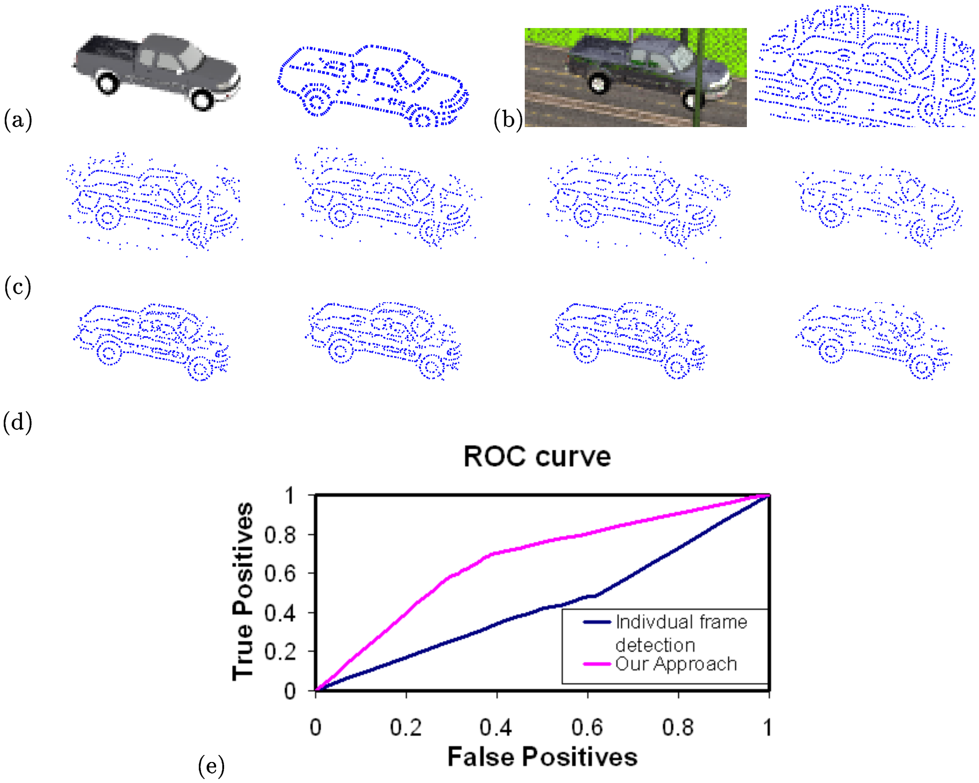


Fig. 11. This figure illustrates the experiment carried out on synthetic video to evaluate our approach quantatively. (a) image and edge map of the object in simple setting, (b) cropped image and edge map of the object in more realistic conditions, (c) Foreground edge-detection with increasing threshold (left to right), (d) Enriched edgemaps corresponding to edge-maps (c) and (e) ROC curve for our approach and compared to single detections.

In addition to above results we also provide quantative evaluation for our approach. The task is to compare edge-map obtained from our approach to the underlying real edge-maps. One can obtain some sort of ground truth or the underlying real edge-map manually by marking edges on each frame of a video. But this would not only be tedious but also would not be accurate. Note that our approach tries to enrich the edge-map which is corrupted by illumination changes, interreflections from surrounding objects, blending of the object into background and occlusion in some cases. Instead, we construct a fairly realistic looking synthetic video and compare the real edge-map of the object as the one estimated when the object is isolated from the complexity introduced by the background. This was done using 3D software POVRAY [8] to render a 3D scene with a vehicle moving in a scene as shown in last row of Figure 10.

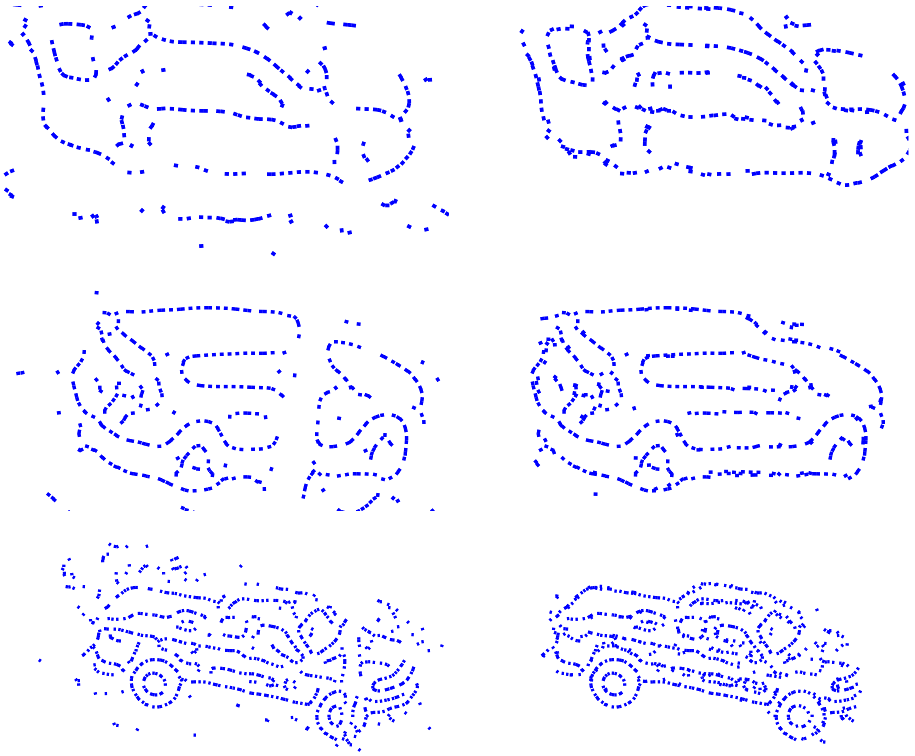


Fig. 12. This figure shows the zoom in on the results from Figures 8, 9, 11. The left column shows the single frame foreground detections and the right column shows the multi-frame composite map. Clearly right column edge-maps are a lot better.

Edge map of the isolated vehicle (first row Figure 10) will be considered the ground truth and we will compare edge-maps obtained by our approach, Figure 11(b), to the ground truth, Figure 11(a). We varied the threshold of the foreground edge detector from the range of 0 to 1. The corresponding foreground edge maps are shown in Figure 11(c). Note as the threshold increases (left to right) the edge-maps get sparse. We ran our algorithm and obtained the enriched edge-maps as shown in Figure 11(d). Next we compared all the edge maps to our ground-truth edge-map, Figure 11(a) and computed false positives and true positives. Next we plotted these values in a ROC curve shown in Figure 11(e). Note clearly the enriched edge-maps have outperformed the original detections.

6 Conclusions and Future Work

Our contribution has been primarily to demonstrate that dynamic or temporal information enhances our detections obtained from a single image as evident from the summary in Figure 12. Some of the drawbacks of our approach are (i) the method breaks down if the object undergoes significant change of viewpoint in the adjacent frames , (ii) the algorithm is computationally expensive as it takes 30 sec on Intel Xeon 3.2GHz processor to process one object (approx. 700 edges). This limits our approach to be applied on objects and not on the whole image. We would like to improve these aspects of the algorithm in future.

Acknowledgements: This research was funded by Lockheed Martin Corporation and NSF grant NSF IIS-0413215.

References

1. Abidi, M.A., Delcroix, C.J.: Analytic fusion of edge maps. In: Proceedings of IEEE Southeast Conference. Volume 2. (April 1989) 739–744
2. Yocky, D.A.: Image merging and data fusion by means of the discrete two-dimensional wavelet transform. Journal of Optical Society of America **12**(9) (1995) 1834–1845
3. Yang, J., Blum, R.S.: Multi-frame image fusion using the expectation-maximization algorithm. In: 8th International Conference on Information Fusion. Volume 1. (July 2005)
4. Jain, V., Kimia, B.B., Mundy, J.L.: Background modeling based on subpixel edges. In: International Conference on Image Processing. (2007) Accepted
5. Chui, H., Rangarajan, A.: A new point matching algorithm for non-rigid registration. Comput. Vis. Image Underst. **89**(2-3) (2003) 114–141
6. Clough, R., Tocher, J.: Finite element stiffness matrices fr analysis of pplates in bending. In: In Proc. of Conference on Matrix Methods in Structural Analysis. (1965)
7. Tamrakar, A., Kimia, B.B.: Combinatorial grouping of edges using geometric consistency in a lagrangian framework. In: Proceedings of IEEE Workshop on Perceptual Organization in Computer Vision, POCV. (2006) 189–197
8. POVRAY: Persistence of vision pty. ltd, persistence of vision raytrace (2004)