2009 13th International Machine Vision and Image Processing Conference

Feature-Assisted Sparse to Dense Motion Estimation using Geodesic Distances

Dan Ring and François Pitié
Sigmedia, Dept. of Electronic and Electrical Engineering
Trinity College, Dublin
dan@unworkable.org, fpitie@mee.tcd.ie

***Abstract*—Large motion displacements in image sequences are still a problem for most motion estimation techniques. Progress in feature matching allows to establish robust correspondences between images for a sparse set of points. Recent works have attempted to use this sparse information to guide the dense motion field estimation. We propose to achieve this in an extended motion estimation framework, which integrates information about the geodesic distance to the sparse features. Results show that by considering a handful of these feature matches, the geodesic distance is able to propagate the information efficiently.**

***Keywords*-Local features Motion vector estimation Large displacement Geodesic distance candidate selection**

I. Introduction

Modern motion estimation algorithms perform well on images with relatively low inter-frame displacements, but exhibit problems when the motion becomes large. Feature-assisted motion estimation aims to combine the ability of feature correspondences to handle large displacements, with the dense, sub-pixel accuracy of typical motion estimation algorithms.

Energy minimisation schemes such as graph-cuts work well when the number of possible labels is low. For example, binary segmentation (2 labels), or depth estimation (256 labels [1]). In motion estimation however, the number of possible displacement vectors for each pixel site is large. To reduce the number of candidates, most algorithms will limit the number of candidates to a small region around the current site, and use some form of coarse-to-fine, multi-resolution image pyramid framework [2], [3], [4], [5]. We propose to use the ability of sparse feature matches to identify both large and small motion between a pair of images to prune the set of displacement vectors. This allows a more concise set of motion displacements to be found using existing energy minimisation techniques without the need of a multi-resolution framework.

The field of motion estimation already shares a long history with feature points [6], [7], [8], [9], [10], [11], [12], [13]. Of particular relevance to this paper is the work of Wills et al. [14]. Their work begins by putatively matching feature points between an image pair, then using these matches to fit planar motion models (homographies) to selections of features using RANSAC. Pixels are then assigned labels corresponding to the finite set of motion models by minimising a suitable energy function with graph-cuts [15]. The main problem with this estimation framework is that the candidate motion vectors are limited to the number of fit motion models. Apparent motions that are not sufficiently planar are not considered as candidates. Additionally, the accuracy of the dense per-pixel motion is based on how well each motion model is fit.

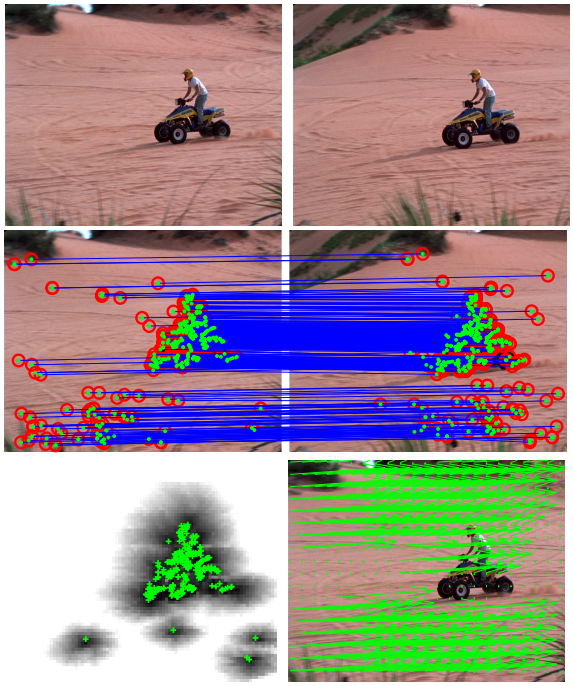


Figure 1: This paper proposes to use feature-matches between a pair of images (top & middle) to identify large motion displacements and geodesic proximity to correct matches (bottom, left), to perform sparse to dense motion vector estimation (bottom, right).

Other recent work by Smith et al. [16] demonstrates using feature correspondences directly to identify viable motion candidates. The dense optical flow is found by simply selecting the candidate with the minimum displaced-frame-difference (DFD). Lombaert et al. [17] propose to extend the standard motion estimation energy model to include the spatial proximity of the sparse features in the likelihood model, encouraging local dense vectors to have similar

978-0-7695-3796-2/09 $26.00 © 2009 IEEE
DOI 10.1109/IMVIP.2009.9

displacements as their nearby sparse feature matches. Note however that, contrary to Wills et al. or Smith et al., the values of the motion vectors are not limited to the candidates given by the feature matches, and can take any value (+/- 15 pixels both in the x and y direction). Feature matches are only used to bias the motion field near the landmark features.

Contributions: In this work, we propose to extend the framework of Lombaert et al. [17] to include similar motion candidate selection schemes to the one of Smith et al. [16] or Wills et al. [14]. We argue in the following sections that limiting the range of possible motion vectors from thousands of candidates to a few ones is both effective at reducing the computational complexity and also at simplifying the complexity of the model, mitigating the need of complex motion priors.

The second contribution of this paper is to refine the notion of spatial proximity to features of Lombaert et al. [17]. Choosing which nearby feature is the most useful is key to the problem. In [17], the distance to the nearest sparse feature is given by the L^2 distance. We propose instead to use a geodesic distance. The idea is to account for the image topology and make better use of the sparseness of the feature set.

II. Feature-Assisted Motion Estimation

Following the work of Lombaert et al. [17], we extend the standard framework for motion estimation to also include sparse motion vector candidates.

Denote as I_1 and I_2 the two frames under consideration. At each pixel site p the motion vector is denoted as d_p. The motion field $(d_p)_p$ is modeled as a Markov Random Field (MRF). The maximum a posteriori solution can be found by minimizing the following energy:

$$E = \sum_p U_p(d_p) + \sum_{p,q \in \mathcal{N}_p} V_{p,q}(d_p, d_q) + \sum_p W(p, d_p) \quad (1)$$

The first term $U_p(d_p)$ corresponds to the likelihood that d_p is the motion vector at pixel p. In this paper, this energy is set to be the displaced frame difference (DFD) of a 8×8 neighbouring image block.

The second term $V_{p,q}(d_p, d_q)$ expresses the spatial smoothness of the motion field. The pixel pair interactions are defined between neighbouring pixels ($q \in \mathcal{N}_p$) to favour a smooth motion field:

$$V_{p,q}(d_p, d_q) = \lambda \|d_p - d_q\| \quad (2)$$

The amount of smoothness is controlled by λ. These first two terms are standard components of the motion estimation model.

The last term $W(p, d_p)$ is a generalisation of the energy term introduced by Lombaert et al. [17]. Consider that we sampled a sparse set $\mathcal{K}$ of features positions $\{p_k\}_{k \in \mathcal{K}}$ with candidate motion vectors $\{d_k\}_k$. The energy $W(d_p, p)$ is a prior term which favours the motion to be d_k near the feature point p_k. For instance, in Lombaert et al., the energy is set as follows:

$$W(p, d_p) = \mu \sum_{k \in \mathcal{K}} \frac{1}{\|p - p_k\|^2} \delta(d_p \neq d_k) \quad (3)$$

where $\delta(d_p \neq d_k)$ is 1 for $d_p \neq d_k$ and 0 otherwise, and the variable μ is used to weight the contribution W in Eq.(1).

The minimisation of the overall energy of Eq.(1) can be done using typical MRF optimisations techniques, such as Graph-Cuts or Belief Propagation. In this paper we considered the α-expansion algorithm from [15].

A. Motion Candidate Selection

Restricting the range of possible displacements is required to reduce the computational load. In the approach of Lombaert et. al. [17], although the motion range is limited to plus or minus 15 pixels, 961 possible motion vectors exist for each pixel. By choosing instead a dozen candidates, optimization techniques such as Graph-Cuts become tractable.

Another advantage of using less candidates is that the situation becomes less ambiguous, notably on flat regions where many motion vectors could lead to a low DFD. By restricting the range of motion candidates, chances are that only the correct motion candidate will result in a low DFD. Since correct motion vectors are more distinctive, the simple prior model proposed in Eq. 1 is better able to propagate information into flat areas.

In this work, the features are sampled using Harris-Laplace keypoint detector [18] and the associated feature descriptors are the SIFT descriptors [19] (see Figure I). A match in the other image is obtained by finding the feature with the most similar SIFT descriptor vector. As it is assumed that the object will not undergo dramatic changes in rotation and scale between images, incorrect matches can be identified and rejected if the orientation of the descriptors differs by more than 60°, or if the difference in scale factors differs by more than 1.5. These canonical rotation and scale variables are calculated as part of the feature detection process, using the algorithms described by Lowe [19] and Mikolajczyk [18] for rotation and scale respectively. Matches are also rejected if the L^2 distance between the matched SIFT pair of descriptors exceeds 0.4[1].

Candidate Pruning: After feature matching, the number of matches can range in the thousands, depending on the nature of the images. Each feature pair match gives a motion candidate. To reduce the number of candidates to a handful, the motion candidates are clustered using the MeanShift algorithm [20]. Clustering the motion vectors is a similar idea to the one of Willis *et al.* [14]. In [14], the RANSAC procedure is iterated to cluster candidates into different planar motion models (*i.e.* homographies). The difference is

[1]This is assuming that the descriptors are normalised according to Lowe's [19] implementation, such that the L^2 distance of each descriptor is 1.

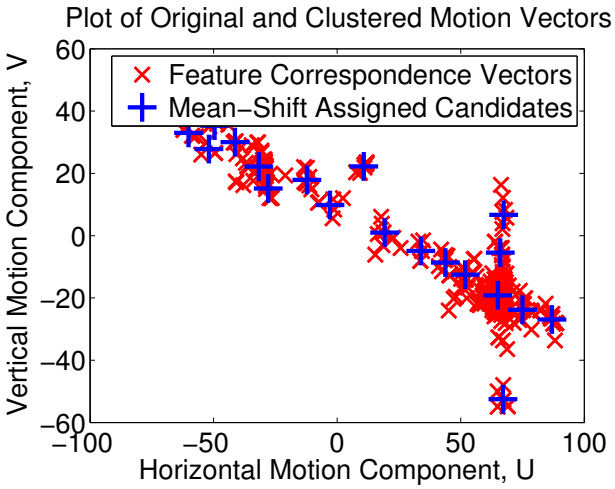


Figure 2: Example of motion vector candidate clustering.

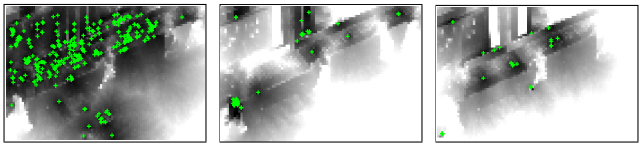
that we restrict the motion model to be translational and that our clustering is directly based on kernel density estimation instead of RANSAC. Our motivation is that simple models yield fewer outliers. In Figure 2, motion vectors found for each feature match (red) and the centres of each motion cluster (blue) are shown.

The cluster centres serve as the pool of candidates that we will use. The mean shift is run for a bandwidth of 10 pixel and clusters that have less than 5 matches are discarded. Thus we end up with a set of features $\{p_k\}_{k\in K}$ which are associated with motion vector candidates $\{d_k\}_{k\in K}$. Note that the motion candidates to be used now are not given by the original matches, but instead by the cluster centres, thereby greatly reducing the overall number of candidates. Depending on the complexity of the apparent motion, the number of centroids that yield good results will range between 2 and 10.

Once the limited set of candidates have been identified, they can then be refined if desired. Additional candidates are randomly sampled in the vicinity of existing candidates, effectively adding "jitter" to the candidate motion vectors. For example, for each cluster centroid, 10 additional random motion vectors similar to the centroid vector can be added. This allows for more subtle variations in the motion field, producing more accurate correspondences. The impact of the candidate sampling strategy is discussed further in the results section.

B. Geodesic Distance To Features

The sparse to dense strategy relies on propagating the sparse information to the pixel level. Since we are confident that the set of candidate motion vectors is representative of the entire motion field, it only remains to find for each pixel a feature that follows the same motion. The key of the problem is thus to assign a feature to each pixel.



Candidate 1 Candidate 2 Candidate 3

Figure 3: Geodesic Distance Maps for the three Motion Candidates. Locations in bright regions are far away from features (geen) with the same motion candidate. Points in dark regions have a feature with the same motion nearby.

Intuitively, the nearest feature is likely to present the correct motion. The estimated feature sets are however in general too sparse to guarantee this. A Bayesian approach must then be taken. In Lombaert et al. [17], the term with W acts as an additional prior based on the L^2 distance to the features. We argue that a more powerful approach is to consider a geodesic distance between the pixel and the features.

The geodesic distance has been used in images for some time now. The distance takes into account the topology of the image. Points that are within the same flat region are close to each other, whereas points that are separated by strong image gradients are far away. The mathematical expression for the geodesic distance below is quite complex, but its implementation using Fast Marching techniques [21] is very simple.

$$D_{Geo}(p,q) = \min_{\Gamma\in\mathcal{P}(p,q)} \int_0^1 \sqrt{\|\Gamma'(s)\|^2 + \mu^2 \left(\nabla I \cdot \frac{\Gamma'(s)}{\|\Gamma'(s)\|}\right)^2} \tag{4}$$

The proposed prior energy is thus as follows:

$$W(p,d_p) = \mu \min_{k\in\mathcal{K}|d_k=d_p} D_{Geo}(p,p_k) \tag{5}$$

The distance maps for three different candidate vectors are shown in Figure 3. For a candidate vector d_p, points p that are on dark regions are close to features p_k that present the same motion $d_k = d_p$. Points in bright regions are far away from these features. Note that the topology is now closely following the contours of the image objects and thus more meaningful than when using the L^2 distance.

III. Results

To illustrate the effects of the various parts of the proposed motion estimation, the results of three experiments are presented. The original figures used throughout the experiments are shown in Figure 4. Masks of regions that exist in the first image, and are occluded in the second, were manually created and shown in red. Motion estimation in these regions will be incorrect. The first two experiments compare the number of candidate features and the effect of refined candidates against performance. The interest of the geodesic distance is then measured. The last experiment compares

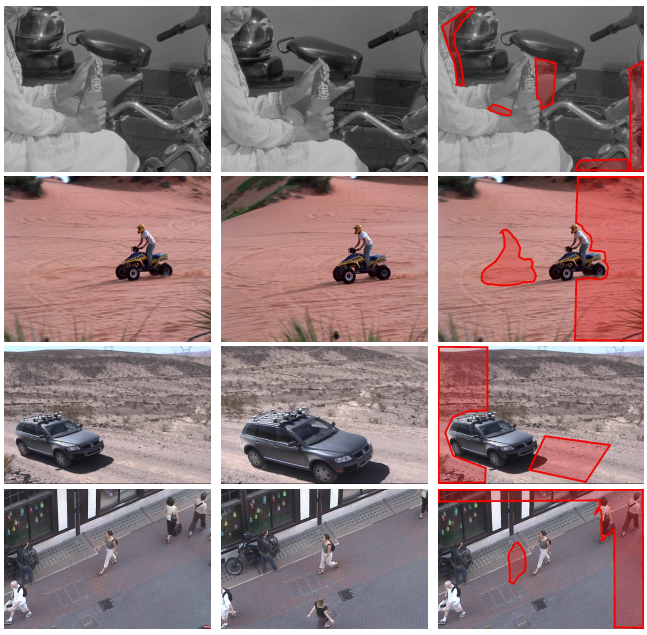


Figure 4: The frames chosen for the experiments, from top to bottom, “Bike”, “Quadbike”, “SUV”, “Walking” scenes. Each pair of images are taken 20 frames apart. The regions highlighted in red are occluded regions, *i.e.* regions that are visible in the first frame, and not visible in the second frame.

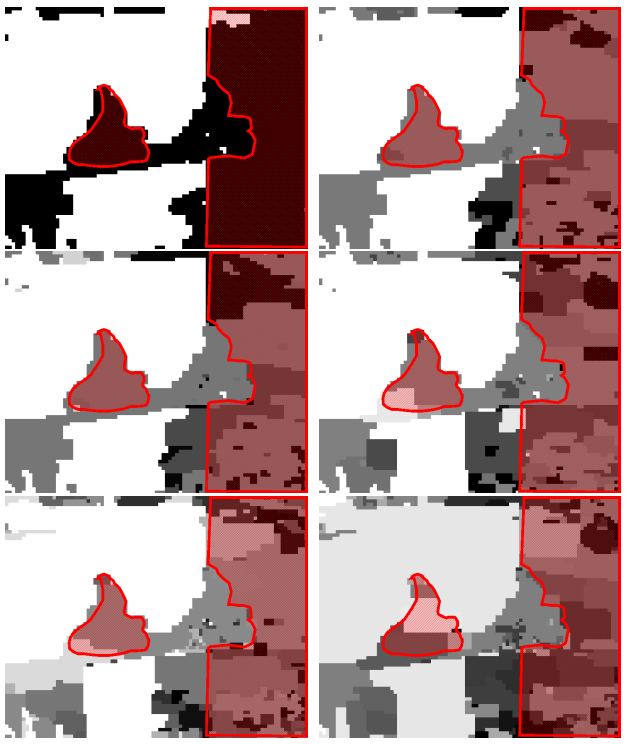


Figure 5: Labelling of “Quad-bike” sequence using (from top-left to bottom-right) 2, 5, 7, 10, 20 & 30 motion candidates. Similar gray-values indicate similar direction & magnitude. This sequence has two main motion components, that of the background moving right, and the stationary rider. Notice how the motion labelling becomes noisier as the number of candidate displacements increases.

our algorithm to a version of the Horn and Schunck motion estimator modified to allow for large displacement.

Candidate Sparseness: The results in Figure 5 show how limiting the number of total motion candidates close to the number of apparent motions improves results. By allowing more models, the likelihood of assigning an incorrect motion label using poor DFD information increases.

As shown in Figure 7, adding more candidates can however be useful if the extra candidates are randomly sampled in the vicinity of the existing candidates. The new candidates allows more subtle variations in the motion field, producing more accurate correspondences.

Feature Proximity Distances: In Figure 6, we compare the results of motion labels obtained using a) only the DFD (as used in traditional motion estimators), b) only the L^2 distance to nearby feature matches (used by Lombaert et al.), c) the proposed geodesic distance, and d) the geodesic distance and DFD combined. The DFD data is very noisy, and not very useful alone in these cases. The L^2 distance alone provides reasonable results, giving a smoother labelling than the DFD. The geodesic distance performs very well, segmenting the motion fields by encouraging large smooth regions. The geodesic distance combined with the DFD introduces more detailed information into the labelling (d).

Comparison with Gradient Based Techniques: Lastly, in Figure 8 we compare our results with a hybrid motion estimation of Horn & Schunck and Lucas - Kanade motion estimation (HSLK). The HSLK algorithm uses a multi-grid, gradient based technique to solve the first terms of Eq.(1). The results show that for cases where the traditional motion estimator works poorly (“Quad-bike” and “Walking”), our algorithm performs reasonably, clearly identifying the dominant motions. For the “Bike” sequence, where the HSLK performs well, our algorithm presents a similar, but quantised version. This is expected from using quantised candidate motion vectors.

IV. Discussion & Conclusion

The difficulty in sparse to dense strategies is to find how to select useful matches and how these matches should influence the dense motion flow. We address these two aspects of the problem by offering a simple, yet effective feature sampling strategy and proposing the geodesic distance to efficiently influence whole regions of the picture.

Results from our experiments show that motion candidates should be kept small, to both allow for global optimization techniques but also because it improves the quality of the

results. We have also found that the geodesic distance was a powerful tool on its own, able to propagate the information over longer distances than simply relying on the spatial smoothness alone.

The results achieved with our method are still limited by the quantization of the motion vectors but the important point is that the method is much more robust than gradient-based techniques such as Horn & Schunck and Lucas & Kanade's motion estimations. The generated motion fields are quite sensible and able to cope with large displacements.

REFERENCES

[1] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *Computer Vision, IEEE International Conference on*, vol. 2, 2001, pp. 508–515.

[2] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[3] J. Konrad and E. Dubois, "Bayesian estimation of motion vector fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 9, pp. 910–927, 1992.

[4] A. C. Kokaram, *Motion Picture Restoration: Digital Algorithms for Artefact Suppression in Degraded Motion Picture Film and Video*. Springer Verlag, 1998.

[5] M. L. C. W. Y. Zhang, "Multi-resolution optical flow tracking algorithm based on multi-scale harris corner points feature," in *Chinese Control and Decision Conference*, 2008, pp. 5287–5291.

[6] C. Harris, "Determination of ego-motion from matched points," in *Proceedings of 3rd Alvey Vision Confernce*, September 1987, pp. 189–192.

[7] H. Wang and M. Brady, "Real-time corner detection algorithm for motion estimation," *Image and Vision Computing*, vol. 13, no. 9, pp. 695 – 703, 1995.

[8] C. Tomasi and T. Kanade, "Detection and tracking of point features," Carnegie Mellon University, Tech. Rep. CMU-CS-91-132, April 1991.

[9] J. Shi and C. Tomasi, "Good features to track," in *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1994, pp. 593–600.

[10] R. Chellappa, G. Qian, and S. Srinivasan, "Structure from motion: sparse versus dense correspondence methods," in *Image Processing, IEEE International Conference on*, 1999, pp. 492–499.

[11] P. Torr and A. Zisserman, "Feature based methods for structure and motion estimation," in *Vision Algorithms: Theory and Practice, number 1883 in LNCS*. Springer-Verlag, 2000, pp. 278–295.

[12] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[13] F. Dellaert, S. Seitz, C. Thorpe, and S. Thrun, "Structure from motion without correspondence," in *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, June 2000.

[14] J. Wills, S. Agarwal, and S. Belongie, "A feature-based approach for dense segmentation and estimation of large disparity motion," *International Journal of Computer Vision*, vol. 68, no. 2, pp. 125–143, 2006.

[15] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts." *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 11, pp. 1222–1239, 2001.

[16] T. Smith, D. Redmill, C. Canagarajah, and D. Bull, "A framework for dense optical flow from multiple sparse hypotheses," in *Image Processing, IEEE International Conference on*, 2008, pp. 837–840.

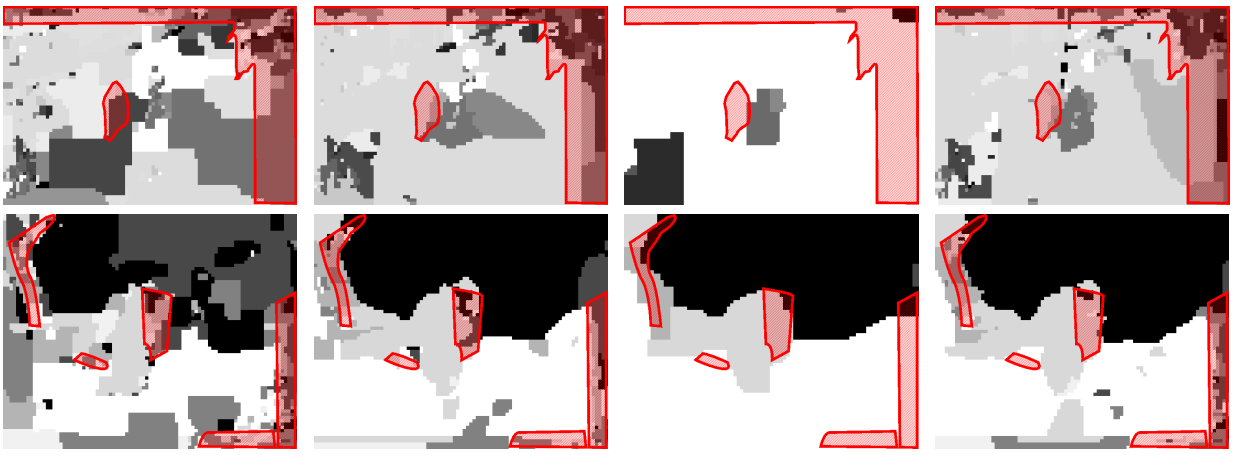
[17] H. Lombaert, Y. Sun, and F. Cheriet, "Landmark-based non-rigid registration via graph cuts," in *International Conference on Image Analysis and Recognition*. Springer-Verlag, August 2007, pp. 166–175.

[18] K. Mikolajczyk, B. Leibe, and B. Schiele, "Local features for object class recognition," in *Image Processing, IEEE International Conference on*, 2005, pp. 1792–1799.

[19] D. G. Lowe, "Distinctive image features from scale-invariant keypoints." *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, November 2004.

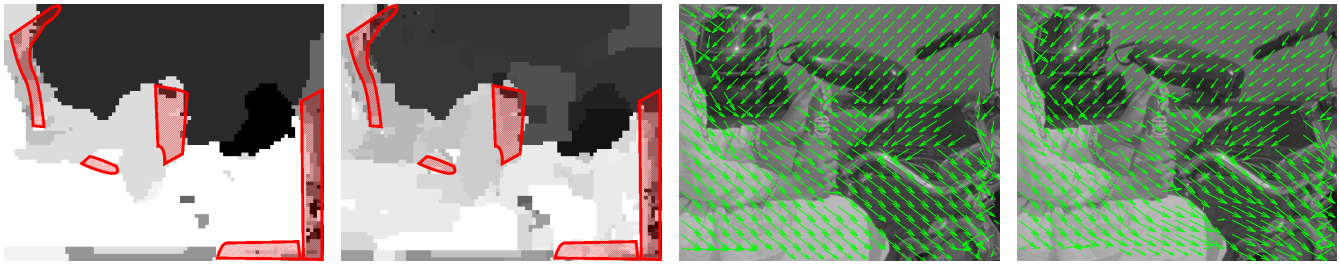
[20] D. Comaniciu and P. Meer, "Mean shift analysis and applications," in *Image Processing, IEEE International Conference on*, 1999, pp. 1197–1203.

[21] L. Yatziv, A. Bartesaghi, and G. Sapiro, "O(n) implementation of the fast marching algorithm," *Journal of Computational Physics*, vol. 212, no. 2, pp. 393–399, 2006.



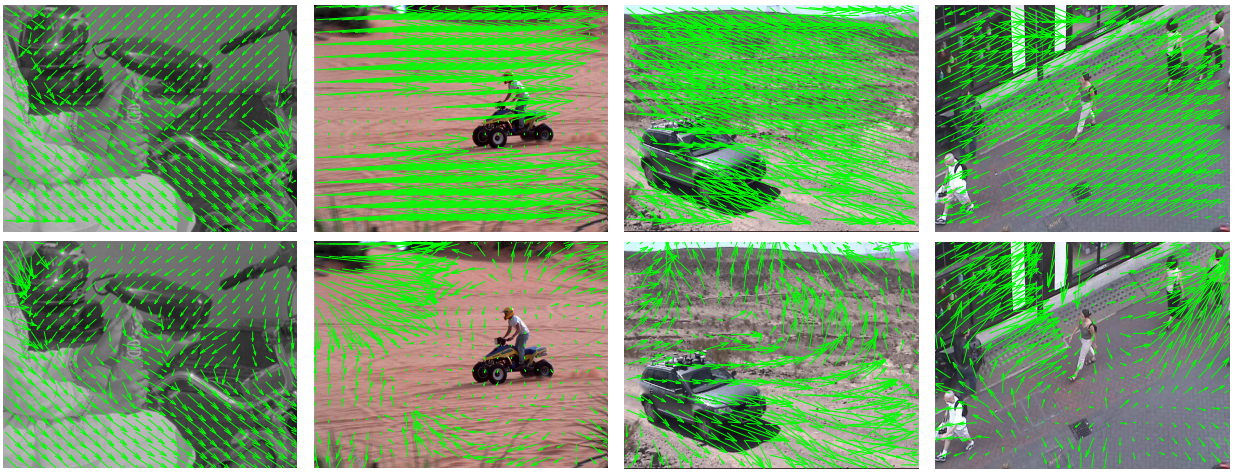
(a) DFD (b) L^2 Distance (c) Geodesic (d) Geodesic & DFD

Figure 6: Label fields generated using (left to right) the DFD alone, the L^2 distance, the proposed geodesic distance, and the use of the geodesic distance and DFD together, for the "Walking" (top) and "Bike" (bottom) scenes.



(a) Orig. Labelling (b) Refined Labelling (c) Orig. Motion Field (d) Refined Motion Field

Figure 7: Example of refining candidate displacements by adding random, "jitter" candidates in the neighbourhood of the existing candidates. From left to right, the motion labels before and following candidate refinement, and the motion field before and after refinement, for the "Bike" scene. The number of candidates between unrefined and refined were increased five-fold, i.e. from 6 to 30.



(a) "Bike" (b) "Quad-bike" (c) "SUV" (d) "Walking"

Figure 8: Results of comparisons between our proposed detector (top) and a hybrid between Horn and Schunck and Lucas - Kanade motion estimation (HSLK) algorithm (bottom) for various scenes.