ROBUST SPATIO-TEMPORAL FEATURE TRACKING

Matthias Heinrichs*, Olaf Hellwich and Volker Rodehorst

Computer Vision & Remote Sensing, Berlin University of Technology, Franklinstr. 28/29, FR 3-1,
D-10587 Berlin, Germany – (matzeh, hellwich, vr)@cs.tu-berlin.de

KEY WORDS: Visual marker-less tracking, spatio-temporal constraints, trifocal geometry, guided matching, SIFT descriptor, interest points, video sequence

ABSTRACT:

Simultaneous tracking of features acquired by multiple video cameras mounted on a rig opens new possibilities for ego-motion estimation and 3D scene modeling. In this paper we propose a novel approach of tracking three video streams at once. The color image features are detected using interest operators and described with SIFT. Since standard tracking techniques perform outlier detection only according to relative orientation between temporal image pairs and hence suffer from outliers which cannot be identified by the epipolar constraints, we improve the outlier detection using temporal and spatial trifocal constraints. Furthermore, these spatio-temporal constraints allow the system to perform a guided matching, which increases the number of tracked features.

1. INTRODUCTION

Tracking of sparse features throughout an image sequence is required for many applications such as ego-motion estimation, 3D scene reconstruction and augmented reality. Vision-based trackers are popular due to their accuracy, flexibility and convenient handling. Many approaches are based on fiducials or specific markers, such as the ARToolKit (Kato & Billinghurst, 1999). The direct use of scene features without any visual markers simplifies and generalizes the tracking process for many applications. Nevertheless, tracking of natural scene features in unprepared environments is still a challenge.

An important application of tracking is the estimation of camera motion. The reconstructed sensor path is intuitively better, if the input is free of outliers, the localization is highly accurate and the features are numerous and well distributed. Many feature tracking techniques are based on the motion smoothness constraint to filter outliers. This constraint is very restrictive due to the variety of real video sequences with abrupt changes.

Our approach exploits epipolar constraints between successive frames to perform an outlier check. Unfortunately, by using two video images, outliers on the epipolar line cannot be detected. An extension of this technique is the trifocal geometry that describes the relationship of a point triplet over three successive frames.

In case of non-rigid scenes or moving objects the camera path can be stabilized by analyzing multiple synchronized video streams. In this paper we distinguish between temporal tracking of features from one frame to the next and spatial matching of features between different cameras. The integration of tracked and matched features enables new consistency constraints for outlier filtering and guided feature detection. We show how to detect features in one stream with the knowledge of the other streams and ensure consistency and uniqueness over all streams.

This paper is structured as follows. Section 2 starts with a brief description of the used camera system. An overview of different feature detectors and descriptors followed by common tracking procedures are presented in sections 3 and 4. Our proposed tracking technique for filtering and guided matching is described in section 5. Some experiments and their results are evaluated in section 6. Finally, conclusions and future directions of research are discussed.

2. SYSTEM OVERVIEW

Our exemplary video camera system for image acquisition consists of three 5-megapixel CCD sensors mounted on a hand-held rig (see Figure 1). The proposed tracking method does not require a specific orientation (e.g. stereo normal case) or calibration of the cameras. The configuration should ensure an image overlapping area of at least 60 percent. Therefore, the horizontal and vertical base lengths can be adapted to the size and distance of the observed scene. The video cameras are synchronized and capture with a frame rate of 16 Hz.

Figure 1. Mobile trifocal camera system

3. FEATURE DETECTION AND DESCRIPTION

In recent years, there has been a growing interest in feature detection algorithms (Mikolajczyk et al., 2005). Based on the established interest point operator (Förstner, 1994) and corner detectors (Harris & Stephens, 1988), new Harris-affine and Hessian-affine detectors (Mikolajczyk & Schmid, 2004) were invented. Additionally, salient regions (Kadir et al., 2004), maximally stable extremal regions MSER (Matas et al., 2002) and intensity extrema-based region detectors IBR or edge-based region detectors EBR (Tuytelaars & Van Gool, 2004) were proposed.

Most popular is the scale invariant feature transform SIFT (Lowe, 2004) that can be separated in two parts: The feature localization and the feature description part. SIFT identifies distinctive invariant keypoints as local extrema of the difference-of-Gaussian (DoG) images across scales. However, the localization accuracy in scale-space is weaker than that of interest point operators (Rodehorst & Koschan, 2006). Therefore, we exchanged the SIFT localization technique with the Förstner operator in the original image scale. The resulting interest points are not invariant to scale anymore. However, this disadvantage does not limit video tracking applications, because the tracked features do not significantly change their scale in dense image sequences or between the images of cameras on the rig.

3.1 Interest Points

We use adjustable continuous filters to determine the magnitude and the direction of image intensity changes following (Canny, 1986). The spatial derivative of an image function f in x-direction is calculated by convolution with the gradient-of-Gaussian (GoG)

$$f_x(x,y) = f(x,y) * -\frac{x}{2\pi\sigma^4} \cdot \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \quad (1)$$

where the standard deviation σ defines the influence area. The first partial derivative in y-direction can be obtained in a similar way. Interest points are identified by use of the autocorrelation function. We extend the structure tensor **A** to color images (Rodehorst & Koschan, 2006). The partial derivatives are calculated for each RGB color channel on the smoothed image with the natural scale σ and then summed over a Gaussian window G using an artificial scale σ_2 with

$$\mathbf{A}(x,y) = G_{\sigma_2} * \begin{bmatrix} r_x^2 + g_x^2 + b_x^2 & r_x r_y + g_x g_y + b_x b_y \\ r_x r_y + g_x g_y + b_x b_y & r_y^2 + g_y^2 + b_y^2 \end{bmatrix} \quad (2)$$

where the indices of the influence area were omitted for simplicity. The two-dimensional convolution kernels in equation 1 and 2 can be separated into two consecutive one-dimensional convolution operations, one on the image rows and one on the columns. Thus, the cost for a $n \times n$ filter mask with n^2 multiplications and n^2-1 additions reduce to $2n$ multiplications and $2n$-2 additions for each image pixel. Förstner (Förstner, 1994) analyzes the eigenvalues of the inverse of **A** that define axes of an error ellipse. Salient points are represented by small circular ellipses that can be computed from

$$w = \frac{\det(\mathbf{A})}{\mathrm{trace}(\mathbf{A})} \quad \text{and} \quad q = \frac{4 \cdot \det(\mathbf{A})}{\mathrm{trace}(\mathbf{A})^2} \quad (3)$$

where w describes the size and q the roundness of the ellipses. We improve the sub-pixel position for point-like features by paraboloid fitting (Rodehorst & Koschan, 2006). Based on the integer position of a feature with maximum w its direct neighborhood is normalized and fitted with a bi-squared function

$$w(x,y) = ax^2 + by^2 + cxy + dx + ey + f \quad (4)$$

The maximum of the paraboloid defines the sub-pixel position

$$x_s = x + \frac{(2bd - ce)}{(c^2 - 4ab)} \quad \text{and} \quad y_s = y + \frac{(2ae - cd)}{(c^2 - 4ab)} \quad (5)$$

of the interest point.

3.2 SIFT Descriptor

These image features can be characterized using the SIFT descriptor (Lowe, 2004). It is invariant to image noise, radiometric changes, rotation and minor changes in viewing direction. First, we assign a consistent orientation to each interest point based on local image properties. An orientation histogram with 36 bins covering 360 degrees is formed from image gradients around the feature. The gradients are weighted by a circular Gaussian window with $1.5 \cdot \sigma$. All peaks in the orientation histogram within 80 percent of the highest peak correspond to dominant directions of the local gradients. Therefore, multiple feature descriptions are created at the same location for different orientations. In a succeeding step, gradients of 4×4 positions around the feature point are accumulated in an eight bin gradient histogram, yielding a feature descriptor with 128 elements (see Figure 2).

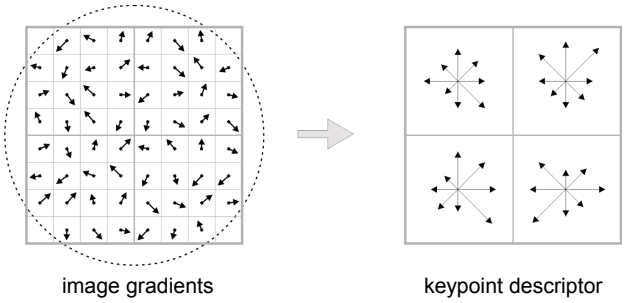

Figure 2. SIFT feature descriptor (Lowe, 2004)

4. ROBUST TRACKING

Now, accurate feature correspondences through a sequence of images must be found. When video streams are acquired at a sufficiently high frequency, frame-to-frame differences are small enough to use optical-flow techniques, such as the popular Kanade-Lucas-Tomasi feature tracker KLT (Shi & Tomasi, 1994).

4.1 KLT Feature Tracker

The iterative algorithm (Birchfield, 2007) computes the optical flow of interest points using image pyramids. During the tracking of features over many frames errors can accumulate. To detect bad matches, the feature in the current frame is compared to the feature in the first frame. Due to perspective distortion, the intensity based consistency check must be performed with an affine mapping (see Figure 3).

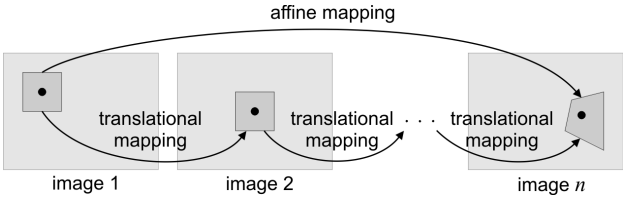

Figure 3. Affine consistency check (Birchfield, 2007)

The OpenCV implementation (Bouguet, 2000) realizes a sparse iterative version of Lucas-Kanade optical flow in pyramids. It is computationally more efficient and finds the correspondences with sub-pixel accuracy.

However, it uses image differences of neighboring pixels

$$f_x(x,y) = \frac{f(x+1,y) - f(x-1,y)}{2} \quad (6)$$

instead of the GoG (see Equation 1) and does not contain the affine consistency check. Another alternative is an implementation of the KLT on a graphics processing unit (GPU), which speeds up the runtime considerably (Sinha et al., 2006).

4.2 SIFT Feature Matching

If frame-to-frame differences are large, feature matching techniques are used instead. Matching SIFT feature descriptors is done by the suggested method of (Lowe, 2004). The cost function between two matching candidates is defined by the Euclidean distance between the describing 128-vectors. These costs are computed for all candidates. The candidate with the lowest distance is accepted, if the ratio of the lowest and the second lowest match is below a given threshold, e.g. 0.6 - 0.8. This approach provides reliable feature correspondences between different views. However, the technique has some disadvantages on repetitive patterns and is computational expensive.

5. TEMPORAL AND SPATIAL CONSTRAINTS

To stabilize the temporal tracking, additional constraints must be used. Epipolar constraints derived from temporal fundamental matrices are very flexible and less restrictive. Unfortunately, they still lead to outliers on epipolar lines. Therefore, tracking consistency is checked over three succeeding frames by use of the trifocal tensor, which can be robustly computed for image triplets by an unfiltered set of matched points. This temporal tensor implies geometric constraints over every pair within this triple. It transfers a pair of corresponding points into the third image, where its position can be verified. Only if a feature triplet fulfils this constraint, it is considered as a possible inlier.

Furthermore, to match a point between the video streams, the same technique for outlier filtering is used. As the orientation of the cameras is fixed, this tensor has to be computed only once and can be reused for every image triplet. If a calibration of the camera rig is available, it can be used for the tensor computation as well. Finally, the spatio-temporal consistency is evaluated (see Figure 4).

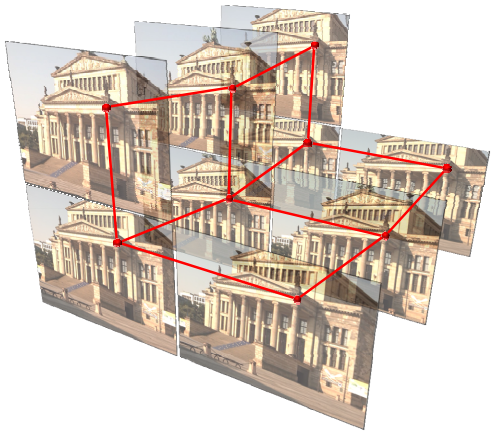

Figure 4. Network of spatio-temporal constraints

If a temporally tracked point is spatially matched between the different streams, the corresponding points of the other streams are checked whether they fulfill their temporal trifocal constraint. This enforces a very tight set of rules, in which outliers are very unlikely to appear. On the other hand, a relatively small number of inliers pass this filter and a lot of inliers are not detected, because of lacking support in the nine frames. Therefore, a guided matching routine checks, if there are features at the predicted positions, which were not detected with the basic SIFT matching strategy.

5.1 Robust Estimation of the Trifocal Tensor

The uncalibrated trifocal tensor can be calculated from at least 6 point correspondences in three images. We use the minimum solver exploiting the Carlson-Weinshall duality (Hartley & Zisserman, 2004). If all points lie on a plane, the tensor can not be computed correctly. Unfortunately, in man-made environments this happens very often due to dominant planes (e.g. facades of buildings). Therefore, a planar homography test over all inliers that are consistent to the trifocal tensor is suggested. To get a robust estimation of the trifocal tensor, a RANSAC approach with an evolutionary strategy called GASAC is used (Rodehorst & Hellwich, 2006). Depending on the noise, a tensor derived from a minimal dataset may lead to quite large epipolar distances among the inliers. Since we implemented no over-determined version of the trifocal-tensor that minimizes the error of all features according to least-squares, a non-linear optimization should be considered as a final step. The complete algorithm can be summarized as follows:

Robust Trifocal Tensor Algorithm

1. Estimate the trifocal tensor with GASAC using 6 points
2. Determine all inliers according to this tensor
3. Test if the inliers define a planar homography?
   - Yes: Remove 2/3 of the planar features from the dataset
     Test if the input has more than 6 points?
     - Yes: Restart from step 1 with the reduced dataset
     - No: Use the planar homography with all inliers
   - No: Trifocal tensor is valid
4. Optional: Non-linear optimization using all inliers

Figure 5. Algorithm outline for a robust trifocal tensor

5.2 Trifocal Filtering

Filtering requires an appropriate error measure, to select the geometrically valid candidates. The trifocal tensor may be used to transfer points from a correspondence in two views to the third view. The projection matrices are directly derived from the trifocal tensor (Hartley & Zisserman, 2004) and allow the computation of all pairs of fundamental matrices:

$$\mathbf{F}^{ij} = \left[\mathbf{P}_j\mathbf{C}_i\right]_\times \mathbf{P}_j\mathbf{P}_i^+ = \left[\mathbf{e}_{ji}\right]_\times \mathbf{P}_j\mathbf{P}_i^+ \quad \text{for } i,j = 1,2,3 \quad i \neq j \quad (7)$$

$\mathbf{P}^+$ is the pseudo-inverse of the projection matrix $\mathbf{P}$, $\mathbf{C}$ the projection center and $[\mathbf{e}_{ji}]_\times$ denotes the skew-symmetric matrix of the epipole of view i arising in view j. Using these fundamental matrices, the Euclidean image distance between the transferred points and the epipolar lines can be calculated. The geometric epipolar error e for a candidate triplet $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$, which are not all epipoles, is:

$$e = \max_{i,j} \left(\frac{\left(\mathbf{x}_j \mathbf{F}^{ij} \mathbf{x}_i\right)^2}{\left(\mathbf{F}^{ij}\mathbf{x}_i\right)_x^2 + \left(\mathbf{F}^{ij}\mathbf{x}_i\right)_y^2 + \left(\mathbf{F}^{ji}\mathbf{x}_j\right)_x^2 + \left(\mathbf{F}^{ji}\mathbf{x}_j\right)_y^2} \right) \quad (8)$$

The maximum guaranties that a bad match of one pair cannot be compensated by an excellent match of another pair. Unfortunately, this measure is numerically not stable in the uncalibrated case and small noise leads to severe misplacement of the transferred point. Therefore, a better quantitative error measurement is suggested. A pair of corresponding points can be triangulated using two projection matrices and reprojected into the third. We found out that the reprojected image position is more stable than using the epipolar point transfer. Both error criteria allow accepting or rejecting a pair of correspondences, where the third correspondence is not found.

5.3 Thresholds

The errors are tested against thresholds, which are computed for every tensor individually and can be obtained automatically. The GASAC-estimation implicitly gives a set of inliers. Computing the maximum epipolar error over all inliers gives a threshold, which guaranties that every new found feature triplet is at least as good as the worst inlier. The threshold for the reprojection error can be derived in a similar way from the maximum distance of all consistent point correspondences to their reprojected image positions. This forms a circular area around every reprojected point, in which the real correspondence must be located. The same technique can be used in case of a planar homography. The threshold is set to the maximum point distance of all consistent point correspondences to their expected positions using the homography transfer.

5.4 Guided Matching

If the trifocal tensor is known, the number of corresponding features can be increased by checking putative candidates with the estimated epipolar geometry (see Figure 6).

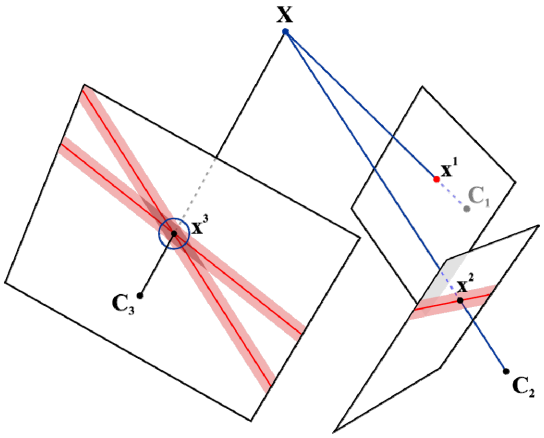

Figure 6. Trifocal geometry

The valid epipolar distances are marked in light red. The intersection area of the search spaces in the third frame limits the feature location to a small rhombic area. A consistent point subset might even contain only one candidate for each frame. Therefore, the basic SIFT matching technique described in Section 3 is extended by a threshold of the descriptor distance to avoid false matching of arbitrary, but geometric valid features. In case of almost linear motion of the camera or object points near to the trifocal plane, which is defined by the three projection centers C_i, the intersecting area is still large. Therefore, the triangulation based technique described in section 5.2 is used to limit the search area in the third view. This area must be computed for every matching candidate in the second view separately. If the trifocal tensor could not be computed because of insufficient camera translation, the planar homography is used to transfer the point to the other views.

5.5 Linked Temporal Matching

Guided matching generates stable spatial features for every image triplet, which are called linked features. In the next step, only these linked features are tracked in time. For further stabilization, the sum of all three descriptor distances to linked feature candidates in subsequent image triplets is used. This results in a set of spatio-temporal features, which usually contains only 1-4 percent outliers. After computing the temporal trifocal tensor from these spatio-temporal features, the linked matching technique is extended by a guided search over linked features similar to (5.4). The result is a linked feature set, which fulfills both, spatial and temporal trifocal constraints.

5.6 Add Virtual Features

An obvious disadvantage of this strict filter is, that losing only one of the nine features rejects the whole set. However, there is a lot of redundancy when the trifocal tensors are available. Each point location can be predicted using two corresponding points with the temporal trifocal tensors or the spatial trifocal tensor. If there is no feature at the predicted image location, one can introduce a new virtual feature. However, to keep in touch with the real world, only one defect on every spatial and temporal trifocal triplet should be tolerated.

5.7 Recognize Lost Features

If a spatio-temporal feature still cannot be tracked in the succeeding frame triplet, its last three descriptors are stored in a stack buffer with a limited size. If the stack is full, the oldest

Spatio-temporal Tracking Algorithm

Initialization:

1. Extract color interest points (4.1) and describe them with SIFT (4.2)
2. Spatial matching on several frames using the ratio of SIFT descriptor distances
3. Spatial tensor estimation (5.1)
4. Compute spatial threshold from inliers (5.3)
5. Guided matching within temporal frames 1 and 2 (5.4)

Main program:

6. For every temporal frame $i \geq 3$:
   a. Guided matching within frame i (5.4)
   b. Track linked features over i-2, i-1, and i (5.5)
   c. Compute three temporal tensors and perform homography check
   d. Compute three temporal thresholds (5.3)
   e. Guided tracking on linked features (5.4)
   f. Add virtual features (5.6)
   g. Recognize lost features (5.7)
7. Fill the retrack stack with all disappeared features

Figure 7. Algorithm outline for the spatio-temporal tracking

features are deleted. This buffer allows the algorithm to track back in time and its size should be set to a reasonable amount of retrackable features, e.g. 5000. If a new spatio-temporal feature is found, it should be tested, if it is very similar to a stored feature in the buffer. As geometric constraints are not applicable to these retracked features, the descriptor distance is tested against a very hard threshold. If successful, the retracked feature is removed from the stack and its ID is reused. The previously described techniques are summarized in Figure 7.

6. EXPERIMENTAL RESULTS

In this section, several experiments with a real outdoor scene have been conducted to demonstrate the advantage of our method. The proposed tracking technique is compared to the standard KLT-implementation for gray value images with affine consistency check disabled (Birchfield, 2007) and an own color variant CKLT using affine matching of every second frame. Both KLT methods use GoG image derivatives and two image pyramid levels. The image sequences were acquired with our trifocal sensor (see Figure 1) while passing Gendarmenmarkt in Berlin (see Figures 8 and 9).

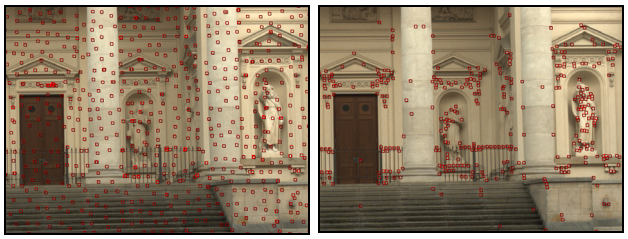

Figure 8. Natural features using KLT (left) and spatio-temporal filtered SIFT features (right)

Figure 9. Image triplet overlaid with permanent tracked features over 10 frames compared to the KLT (top right)

Sequence	Points per Frame			Track length	
	Mean	Min	Max	Mean	Max
KLT for gray value images without affine matching					
C0	1222	892	1402	8.4	105
C1	1230	951	1415	8.7	99
C2	1215	958	1478	11.0	156
CKLT for color images without affine consistency check					
C0	1228	935	1408	8.3	93
C1	1236	973	1386	8.4	100
C2	1225	961	1467	10.9	169
CKLT for color images with affine consistency check					
C0	1211	935	1497	5.2	81
C1	1224	973	1525	5.3	83
C2	1225	961	1529	6.3	96
Proposed method with temporal epipolar constraints					
C0	761	171	1058	8.2	92
C1	753	114	1078	8.2	98
C2	847	322	1111	8.7	124
Proposed method with spatio-temporal constraints					
C0	535	163	752	8.1	95
C1	535	163	752	8.1	95
C2	535	163	752	8.1	95
Proposed spatio-temporal method with virtual features					
C0	652	202	893	10.5	98
C1	643	200	873	10.3	95
C2	652	208	897	10.5	98

Table 1. Tracking results of camera C0, C1 and C2 over 381 frames

The video streams of three cameras C1, C2 and C3 are evaluated over 381 frames each using a resolution of 1384×1038 pixel. The number of tracked features obtained from each technique can be found in Table 1. Additionally, the average and maximum track length of a feature was evaluated.
Comparing the statistical results, the CKLT without affine check tracks more features than the standard KLT, but the mean path length is slightly inferior. The affine consistency check seems to reject many tracked features, but does not stabilize them. The average and maximum track length decreases to 57-58 percent compared to the results without affine checks.
The temporal constrained matcher tracks only 61-69 percent of the average amount of the KLT, while the average and maximum track length are only slightly smaller. This indicates that the KLT has a lot of short tracks, which can be filtered and stabilized by temporal filtering. If spatial filtering between the tracks is used, the number decreases naturally, because only the image overlapping area of all cameras can be tracked, which is approximately 60 percent. Since the three paths are now linked together, the results of the third path C2 will naturally diminish, because there is no link partner in C0 and C1.
We calculated the tracking data with and without virtual features (5.6). The tracks without artificial points have naturally the same values over all three paths. The three paths with virtual features have slightly different values, since artificial features are not evaluated. Without virtual features the path lengths are slightly inferior to the KLT tracker, but the average and maximum track lengths are close to the worst results of the KLT. If virtual features are used, the average path lengths increases compared to the temporal case and even exceeds the KLT in two of three paths. The maximum path length decreases to 93 percent of the KLT path length of track C0.

Since the algorithm can use only 60 percent of the image, this is quite a good result. The average numbers of tracked features decreases to 52 percent in track C1, which is less than the 60 percent overlapping area may suggest. The maximum number of features decreases to an expected amount of 60 percent, compared to the KLT approaches. The minimum number of tracked features of the proposed technique cannot find more than 21 percent of the KLT track. Since the average and maximum track lengths are almost the same or even better, it finds almost every good feature to track and omits a high number of short, weakly or falsely tracked features. This is the main benefit of the proposed technique. The tracked features are highly robust and do not suffer from many outliers, which improve the quality of the output data and while the quantity is only slightly reduced.

7. CONCLUSIONS

In this paper we proposed a novel feature tracking approach by representing color interest points with SIFT descriptors. Furthermore, tracking stability is improved by imposing trifocal filtering, guided matching, virtual point insertion and feature recognition. This approach is based on the trifocal tensor and a geometrically interpretable error measure with an automatically calculated threshold. For the special case of three video streams the tracks are matched and tracked simultaneously. The results from this setup are more stable than tracking every stream independently. A drawback of the proposed tracking approach is that color image features are detected but only gray value structures are matched. Therefore, the SIFT descriptor should be extended to color. Recent investigations of the CSIFT descriptor (Abdel-Hakim & Farag, 2006), Hue descriptor and opponent color derivative descriptor (Weijer & Schmid, 2006) as well as the HSV-SIFT descriptor (Bosch et al, 2006) are highly interesting. These should be incorporated into the proposed method.

ACKNOWLEDGEMENTS

This work was partially supported by grants from the German Research Foundation DFG.

REFERENCES

Abdel-Hakim, A.E. and Farag, A.A., 2006. CSIFT - A SIFT descriptor with color invariant characteristics, *IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 2, pp. 1978-1983.

Birchfield, S., 2007. Kanade-Lucas-Tomasi (KLT) feature tracker, *http://www.ces.clemson.edu/~stb/klt* (accessed 18. Nov. 2007)

Bosch, A., Zisserman, A. and Munoz, X., 2006. Scene classification via pLSA, *Proc. of the European Conf. on Computer Vision*, Vol. 4, pp. 517-530.

Bouguet, J.-Y., 2000. Pyramidal implementation of the Lucas Kanade feature tracker, *http://robots.stanford.edu/cs223b04/algo_tracking.pdf* (accessed 18. Nov. 2007)

Canny, J., 1986. A computational approach to edge detection, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 8, No. 6, pp. 679-698.

Förstner, W., 1994. A framework for low level feature extraction, In: Ecklundh (Eds.): *Proc. European Conf. on Computer Vision*, LNCS 800, Springer, pp. 383-394.

Harris, C. and Stephens, M., 1988. A combined corner and edge detector, *Proc. 4th Alvey Vision Conf.*, pp. 147-151.

Hartley, R. and Zisserman, A., 2004. Multiple view geometry in computer vision, Cambridge University Press, 2. edition, 672 p.

Kadir, T., Zisserman, A. and Brady, M., 2004. An affine invariant salient region detector, *Proc. European Conf. on Computer Vision*, pp. 404-416.

Kato, H. and Billinghurst, M., 1999. Marker tracking and HMD calibration for a video-based augmented reality conferencing system, *Proc. 2nd IEEE and ACM Int. Workshop on Augmented Reality*, pp. 85-94.

Lowe D.G., 2004. Distinctive image features from scale-invariant keypoints, *Int. Journal of Computer Vision*, Vol. 60, No. 2, pp. 91-110.

Matas, J., Chum, O., Urban, M. and Pajdla, T., 2002. Robust wide baseline stereo from maximally stable extremal regions, *British Machine Vision Conf.*, pp. 384-393.

Mikolajczyk, K. and Schmid, C., 2004. Scale & affine invariant interest point detectors, *Int. Journal of Computer Vision*, Vol. 60, No. 1, pp. 63-86.

Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L., 2005. A comparison of affine region detectors, *Int. Journal of Computer Vision*, Vol. 65, No. 1-2, pp. 43-72.

Rodehorst, V. and Hellwich, O., 2006. Genetic Algorithm SAmple Consensus (GASAC) - A parallel strategy for robust parameter estimation, *Int. Workshop "25 Years of RANSAC"* in conjunction with CVPR'06, New York, 8 p.

Rodehorst, V. and Koschan, A., 2006. Comparison and evaluation of feature point detectors, In: Gründig and Altan (Eds.), *Proc. of 5th Turkish-German Joint Geodetic Days*, Berlin, 8 p.

Schmid, C., Mohr, R. and Bauckhage, C., 2000. Evaluation of interest point detectors, *Int. Journal of Computer Vision*, Vol. 37, No. 2, pp. 151-172.

Shi, J. and Tomasi, C., 1994. Good features to track, *Int. Conf. on Computer Vision and Pattern Recognition*, pp. 593-600.

Sinha, S.N., Frahm, J.M., Pollefeys, M. and Genc. Y., 2006. GPU-based video feature tracking and matching, *Workshop on edge computing using new commodity architectures*, Chapel Hill, 2 p.

Tuytelaars, T. and Van Gool, L., 2004. Matching Widely Separated Views based on Affine Invariant Regions, *Int. Journal on Computer Vision*, Vol. 59, No. 1, pp. 61–85.

Weijer, J.v.d. and Schmid, C., 2006. Coloring local feature extraction, *Proc. of the European Conf. on Computer Vision*, Graz, Austria, Vol. 2, pp. 334–348.