# ROBUST ROTATIONAL MOTION ESTIMATION FOR EFFICIENT HEVC COMPRESSION OF 2D AND 3D NAVIGATION VIDEO SEQUENCES

*Dominic Springer*[1] *Franz Simmet*[2] *Dieter Niederkorn*[2] *André Kaup*[1]

[1] Multimedia Communications and Signal Processing, University of Erlangen-Nuremberg
[2] Integration and Test Infotainment, AUDI AG, Ingolstadt

## ABSTRACT

In the context of test automation for automobiles, the compressed video recording of infotainment system components like navigation devices is a required practice. These recordings are then analyzed, archived, and forwarded to the responsible engineering teams. In order to compress navigation video sequences efficiently, the dominant rotational motion must be compensated properly. However, the process of Rotational Motion Estimation (RME) is hindered by the presence of static areas like info boxes and overlay graphics. We analyze this problem and show how to build masks for static areas in order to allow high speed feature transforms to be applied. With the acquired fast and accurate RME, we then demonstrate how to significantly reduce the required bitrate during HEVC encoding of navigation sequences.

***Index Terms***— Rotational Motion Estimation, HEVC, Video Compression, Background Subtraction

## 1. INTRODUCTION

Navigation devices in automobiles have continuously risen in complexity over the recent decade, and so has the need to assert their correct system behavior. For automotive tests, it is common to record the system video output in compressed form, so that analysis and archiving can take place afterward. Typically, the only information available from the system under test is the pixel information at the display output. Information about frame motion can therefore not be retrieved directly. This paper aims at fast and precise estimation of this motion for the purpose of efficient video compression.

As we showed in [1], information about the position of static areas is essential for feature-based motion estimation. In the following, we will extend the state of the art on navigation sequence encoding in three main aspects. Firstly, we evaluate more general ways to gather information about the positions of static areas. This allows us to handle special use cases where H.264/AVC skip mode information may be insufficient for the creation of static area masks. For example, low QPs in combination with slightly transparent static areas may prevent skip mode selection due to non-zero transform coefficients. Secondly, we demonstrate how the masks can be leveraged for stabilizing the process of Rotational Motion Estimation (RME). We compare SIFT [2] and ORB [3] based RME and show how the constrained but real-time capable ORB transform can be used for fast RME on 2D and 3D navigation sequences. Thirdly, we integrate our RME scheme into the HEVC video coding standard [4] and show that the required bitrates can be reduced by up to 9.6%. Note that our scheme is not confined to navigation sequences but applies to any video data with (semitransparent) text or symbol overlays of significant size.

Rotational Motion Estimation (RME) in navigation sequences can be seen as a subtype of global motion estimation, for which authors have suggested both vector-based and pixel-based approaches [5][6], of which the latter is recommended due to its accuracy [7]. To estimate higher order global motion between two frames, it is common to detect and match features on both frames, which can then be used to solve for the correct affine motion parameters [8]. Typically, feature transforms like SIFT [2] or SURF [9] are used for this purpose. However, these come with considerable computational requirements. Computational complexity can be reduced by using KLT feature tracking as in [10], but this introduces an upper bound on detectable rotational motion due to rotational variance. Navigation sequences may show rotations up to 30 degrees between consecutive frames, which makes KLT feature tracking difficult and error prone. In 2011, there has been a promising approach by Rublee et al. [3],

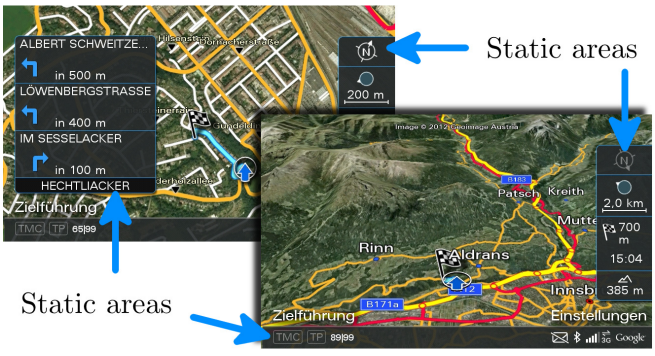



**Fig. 1**. Static areas in Basel2D (left) and Insbruck3D (right).

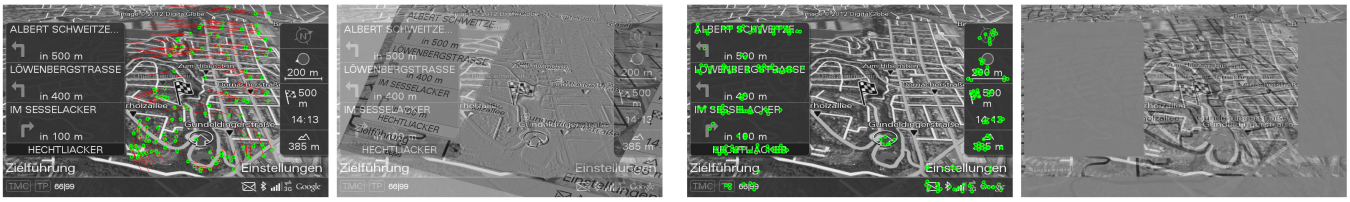



**Fig. 2**. Example of SIFT-based RME on Basel3D. The left image pair depicts a successful RME, where SIFT matching is constrained by hand to the map area. The right image pair shows how RME fails when SIFT is applied to the whole frame area.

introducing the so-called ORB transform for robotic vision. Based on the FAST detector [11] and BRIEF descriptor [12], it allows rotationally invariant feature matching at real-time speeds on standard PC systems. Like SIFT or SURF, ORB cannot be applied to navigation sequences directly: the feature detection and matching process is severely impaired by static areas, since the feature count in static areas as well as their low noise in localized position tend to destabilize common outlier removal schemes like RANSAC [13]. As a result, the estimation of motion parameters of the map area fails.

In order to cope with this problem, we found that Background Subtraction (BGS) techniques allow precise estimations of position and size of static areas. BGS techniques have been intensively studied in the past [14][15][16], most notably for the purpose of video surveillance and object tracking. In the scope of video compression, Glantz et al. showed in [10] that BGS techniques can be used to efficiently identify separate moving objects for sprite-based frame processing using KLT-tracking. In the following, we will show how BGS techniques can be combined with rotational invariant feature transforms such that 1) real-time processing remains feasible and 2) estimations can be carried out in a robust way.

## 2. RME FAILURE ANALYSIS

Before describing our proposed solution for fast and robust Rotational Motion Estimation (RME), it is necessary to analyze in detail why state of the art feature-based parameter estimation shows severe cases of failure for navigation sequences. We found that the rejection of map features as outliers after matching is a frequent behavior, which results in wrongly estimated motion parameters and thus in failing RMEs. Note that outlier removal is a required practice since the matching process is inherently prone to errors. This erroneous removal is due to the fact that feature matches in static areas may very well reach a number close to the number of matches on the map, since map matches are harder to obtain and more prone to false matching. Moreover, non-moving features typically show low noise in their detected positions, and thus influence the RANSAC reprojection error metric.

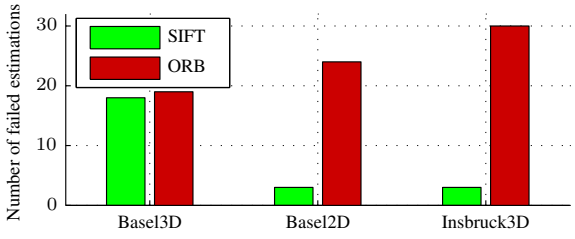



**Fig. 3**. Number of failing RMEs in each sequence when SIFT or ORB is applied on the whole frame. Failures are caused by static areas, which distract the outlier removal process.

In the following, we use three typical 800x480 navigation sequences for demonstration: Basel2D, Basel3D and Insbruck3D. The sequences are real recordings from existing navigation devices and consist of 200 frames each. As Fig. 1 shows, their content covers 2D and 3D map renderings as well as both small and large static area sizes. Fig. 2 gives examples on a successful and a failing RME. In the left image pair, SIFT transform is restricted to the map area by hand masking, excluding any static areas from search. As expected, this allows accurate estimation of the map motion, which is visualized in Fig. 2 by the luminance difference between the current and the warped preceding frame (second image). However, SIFT applied to the whole frame gives many features in static areas, which destabilizes the RANSAC estimation process, effectively resulting in the removal of map features as outliers (see right image pair in Fig. 2). In these cases, the map motion estimation process fails.

2D sequences like Basel2D or 3D sequences with minor static content like Insbruck3D are less prone to this effect. However, this dramatically changes when ORB is used instead of SIFT. This effect is illustrated in Fig. 3. The failing RMEs are due to the strong affinity of ORB to text, which in combination with the hard limit on detector numbers results in an insufficient number of map features. Note that increasing the ORB detector number (we currently use 1500 for our setup) will not help this issue: besides the exponentially increasing matching overhead, map features will still remain a noise-afflicted minority within reasonable detector numbers ($\leq 3000$) and thus are likely to be rejected by RANSAC.

## 3. PROPOSED MASKING SCHEME

The targeted algorithm should be real-time capable and must be able to adapt to the given sequence, since extend and position of map and static areas may change over time. Also, map

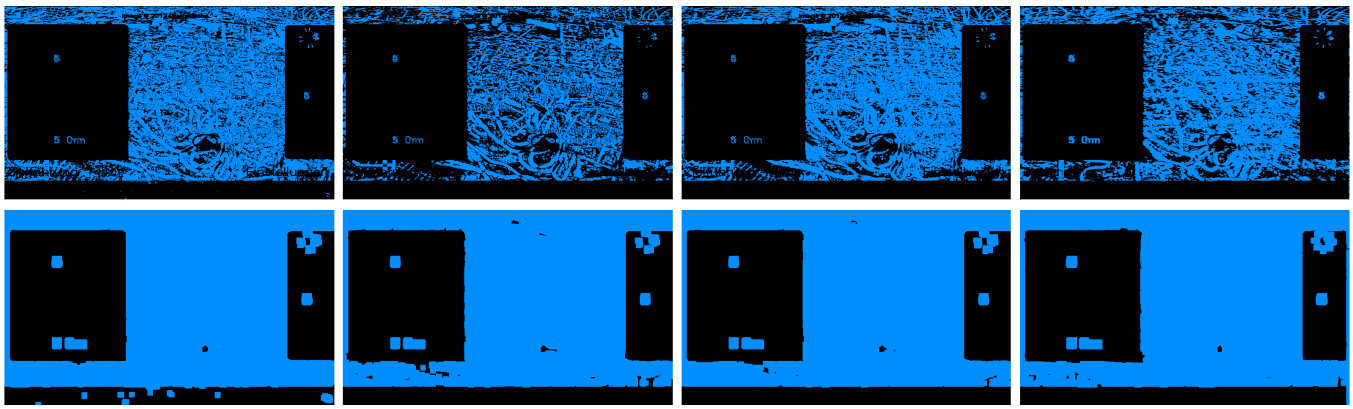
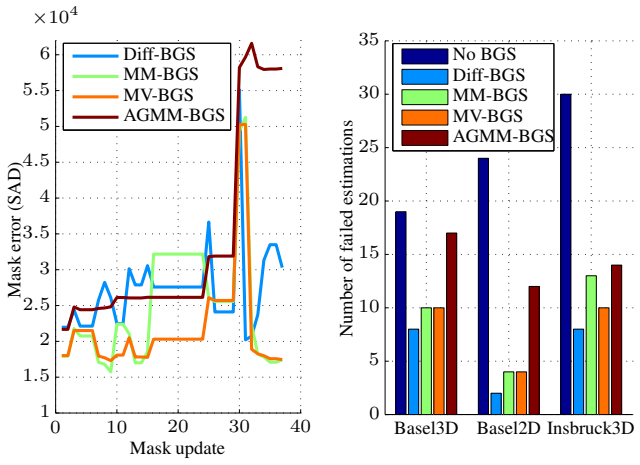


**Fig. 4**. Top row: intermediate mask results for the four tested BGS algorithms Diff-BGS, MM-BGS, MV-BGS and AGMM-BGS (left to right). Bottom row: mask results after morphological hole filling. Blue color indicates estimated map area.

areas may be rendered differently according to the context, which rules out any pattern-based recognition using prior knowledge of the map design. We found that Background Subtraction (BGS) techniques provide both efficient and fast solutions to this problem. In the following, we evaluate four different BGS algorithms: plain frame-by-frame difference (Diff-BGS), moving-mean over the last three frames (MM-BGS), moving variance over the last three frames (MV-BGS) and Adaptive GMM (AGGM-BGS) as representative of a more complex approach [17]. An implementation of those can be found in the open source Bgslibrary [18]. Even though rotation is the dominant motion in navigation sequences, it accounts only for 10-20% of the total frame number and is often interrupted by 1-2 still frames. As a result, MM-BGS, MV-BGS and AGGM-BGS cannot be applied to navigation sequences directly since their moving-window-based processing would result in degenerated background masks. We



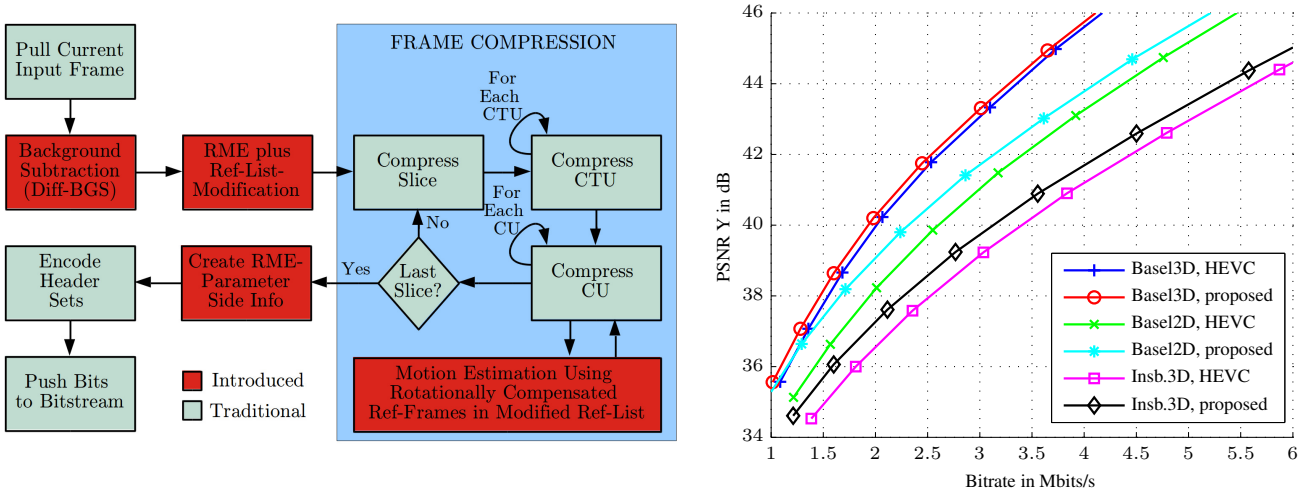

a) b)

**Fig. 5**. BGS evaluation results. a) Mask error with respect to a hand annotated groundtruth mask. b) Number of failed RMEs per sequence when applying different BGS schemes.

therefore use Diff-BGS as a preprocessing step and choose the luminance sum-of-differences (SAD) threshold of successive frames at $10^6$ in order to identify significant motion. A frame is forwarded to MM-BGS, MV-BGS and AGGM-BGS only when motion is found to be significant.

The acquired masks need further postprocessing in order to be used for RME: we apply morphological hole filling with a 13x13 sized rectangular kernel. Fig. 4 depicts example masks for each of the four tested BGS algorithms. Note that static areas may still contain moving elements like sliding text or updated symbols. We found that Diff-BGS typically produces less accurate masks, but shows fast recovery when static areas are updated, e.g. when text changes or info boxes slide into or out of view. This is illustrated in Fig. 5a, which shows the mask error for Basel3D in terms of SAD for each of the four algorithms. For illustration purposes, the error is shown only for those time instants where a mask update occurred (i.e. when the frame shows significant motion). We found that even though Diff-BGS produces less accurate masks, it often leads to a more stable ORB-based RME process. In order to demonstrate this, we measure the reprojection error $\epsilon$ of the estimated parameters using groundtruth motion data and identify a failed estimation by checking if $\epsilon > 5$. Fig. 5b gives numbers on this for different BGS algorithms. Interestingly, the fitness of Diff-BGS for RME is caused by the fast recovery from changes in static areas (see mask update 30 in Fig. 5a), which leads to less text areas exposed to ORB detection and matching during RME. Fig. 5b also shows that a Diff-BGS-based scheme is able to cut the number of failed RMEs by 58%, 92% and 73% for Basel3D, Basel2D and Insbruck3D, respectively. Due to this, we choose Diff-BGS for integration into our proposed RME scheme.

In order to compare processing speed of our scheme to the speed of a common SIFT- or SURF-based RME, we implemented all three schemes using the ORB, SIFT and SURF C++ implementations of OpenCV [19]. Since our scheme is based on Diff-BGS for mask creation, we can reuse its SAD




a) b)

**Fig. 6**. HEVC encoder integration. a) Proposed HEVC-based encoder with new components marked in red. b) Comparison of compression efficiency of the proposed HEVC-based encoder against original HEVC encoding.

output to execute hole filling, ORB and RANSAC only in the case of significant motion activity. Combined with the fast execution of ORB, we achieve a single-thread performance of 10.1 fps on a Core2 Quad processor, as compared to 0.3 fps (SIFT) or 1.1 fps (SURF).

## 4. HEVC ENCODING RESULTS

A successful RME is of utmost importance in order to compensate for rotational motion in the scope of video coding. As shown above, a Diff-BGS-based calculation of masks for static areas can significantly reduce the total number of failed RMEs when real-time ORB features are required. In the following, we demonstrate how the proposed Diff-BGS-based stabilization of the RME process directly translates into reduced bitrate requirements for video encoding. We chose the HEVC implementation HM-7.0 [4] for evaluation purposes, but real-time H.264/AVC video encoding solutions like [20] are also feasible. For integration purposes, we configured HEVC with low delay main profile, PPPP GOP size and I-frame starting. In HEVC, we use the reference frame buffer management to provide rotationally compensated reference frames whenever significant motion is detected. The architecture of the modified encoder is illustrated in Fig. 6a. After RME has been carried out using morphologically corrected masks from Diff-BGS, a rotationally compensated reference frame is embedded as second reference frame in reference list 0. The parameters used for this warping step are transmitted to the decoder as uncompressed 18 byte side information on a per-frame basis. This side information overhead is included in Fig. 6b, which gives a comparison between standard HEVC encoding and our proposed encoder in terms of compression efficiency. For Basel2D and Insbruck3D, mean Bjontegaard [21] bitrate savings of 9.6% and 7.5% are achieved, corresponding to a quality gain of 0.7dB and 0.6dB, respectively. The bitrate savings are a direct result of the provided rotationally compensated reference frame, which keeps the encoder from approximating rotational motion by choosing smallest block partitioning. Since we chose Basel3D as an example for a sequence which is hard to estimate even for SIFT, RME failrate is only cut by half and thus RME comes with moderate savings of 3.2% for this sequence. In any case, creation of static area masks is essential for the success of ORB-based RME: skipping the masking process will result in RD-curves falling back to a compression efficiency almost identical to standard HEVC encoding (compare fail rates in Fig. 5b).

## 5. CONCLUSIONS

In this paper, we presented a scheme to perform fast, reliable and precise Rotational Motion Estimation (RME) on navigation sequences. Since real-time feature transforms like ORB [3] or FAST/BRIEF [11][12] are not directly applicable to navigation sequences due to static areas, we combine the feature transform with restricting masks, which we update only in the face of significant frame activity. For mask creation, we evaluated different Background Subtraction (BGS) techniques and found Diff-BGS [18] to provide sufficiently accurate estimations of static areas. Our evaluation shows that the number of failing RMEs can be reduced by up to 92%. The proposed scheme makes ORB applicable to the area of navigation sequence encoding. This eliminates the need to use SIFT [2] or SURF [9] for precise motion estimations while at the same time provides real-time performance with an average of 10.1 frames per second. An integration of the proposed scheme into the HEVC video compression standard demonstrates bitrate savings of up to 9.6%.

## 6. REFERENCES


[1] D. Springer, F. Simmet, D. Niederkorn, and A. Kaup, "Compression of 2D and 3D navigation video sequences using skip mode masking of static areas," in *Proc. PCS*, Krakow, Poland, May 2012, pp. 301–304.

[2] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.

[3] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. ICCV*, Barcelona, Nov 2011.

[4] ITU/ISO/IEC, "HEVC Test Model HM-7.0," http://hevc.hhi.fraunhofer.de/.

[5] K. Y. Wong and C. L. Yip, "An intelligent tropical cyclone eye fix system using motion field analysis," in *Proc. ICTAI*, 2005, pp. 652–656.

[6] M. Haller, A. Krutz, and T. Sikora, "Robust global motion estimation using motion vectors of variable size blocks and automatic motion model selection," in *Proc. ICIP*, Hong Kong, Sept. 2010, pp. 737–740.

[7] M. Haller, A. Krutz, and T. Sikora, "Evaluation of pixel- and motion vector-based global motion estimation for camera motion characterization," in *Proc. WIAMIS*, London, UK, May 2009, pp. 49–52.

[8] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, second edition, 2004.

[9] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," in *Proc. ECCV*, 2006, pp. 404–417.

[10] A. Glantz, A. Krutz, T. Sikora, and P. Nunes, "Automatic MPEG-4 sprite coding - comparison of integrated object segmentation algorithms," *Multimedia Tools and Applications*, vol. 49, no. 3, pp. 483–512, Sep. 2010.

[11] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Proc. ECCV*, May 2006, vol. 1, pp. 430–443.

[12] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *Proc. ECCV*, Crete, Greece, 2010, pp. 778–792.

[13] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," in *Proc. Com. ACM*, June 1981, vol. 24, pp. 381–395.

[14] N. Friedman and S. Russell, "Image segmentation in video sequences: A probabilistic approach," in *Proc. UAI*, 1997, pp. 175–181.

[15] A. M. Elgammal, D. Harwood, and L. S. Davis, "Non-parametric model for background subtraction," in *Proc. ECCV*, 2000, pp. 751–767.

[16] T. Horprasert, D. Harwood, and L. S. Davis, "A robust background subtraction and shadow detection," in *Proc. ACCV*, Queensland, New Zealand, 2000.

[17] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *Proc. ICPR*, Cambridge, England, 2004, pp. 28–31.

[18] A. Sorbal, "Bgslibrary - Background subtraction library," http://code.google.com/p/bgslibrary/.

[19] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[20] x264 Developer Team, "x264 Open Source H.264/AVC encoder," http://x264.nl/.

[21] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," Tech. Rep., ITU-T VCEG Meeting, Austin, Texas, USA, document VCEG-M33, April 2001.