Robust Global Motion Compensation in Presence of Predominant Foreground

Seyed Morteza Safdarnejad
https://www.msu.edu/~safdarne/
Xiaoming Liu
http://www.cse.msu.edu/~liuxm/
Lalita Udpa
http://www.egr.msu.edu/ndel/profile/lalita-udpa

Michigan State University
East Lansing
Michigan, USA

Abstract

Global motion compensation (GMC) removes intentional and unwanted camera motion. GMC is widely applicable for video stitching and, as a pre-processing module, for motion-based video analysis. While state-of-the-art GMC algorithms generally estimate homography satisfactorily between consecutive frames, their performances deteriorate on real-world unconstrained videos, for instance, videos with predominant foreground, e.g., moving objects or human, or uniform background. Since GMC transformation of frames to the global motion-compensated coordinate is done by cascading homographies, failure in GMC of a single frame drastically harms the final result. Thus, we propose a robust GMC, termed RGMC, based on homography estimation using keypoint matches. RGMC first suppresses the foreground impact by clustering the keypoint matches and removing those pertaining to the foreground, as well as erroneous matches. For homography verification, we propose a probabilistic model that combines keypoint matching error, consistency of edges after homograhy transformation, the motion history, and prior camera motion information. Experimental results on the Sports Videos in the Wild, Hollywood2, and HMDB51 datasets demonstrate the superiority of RGMC.

1 Introduction

The objective of global motion compensation (GMC) is to remove *intentional* (due to camera pan/tilt/zoom) and *unwanted* (e.g., due to hand shaking) camera motion. GMC is utilized in applications such as video stitching, or as pre-processing for motion-based video analysis. Effective motion analysis is the gist of many vision problems, e.g., action recognition, video annotation and video surveillance. For instance, in action recognition, motion analysis via dense trajectories has shown superior performance [15, 23, 24]. However, the moving camera often interferes with the motion of human, thus it is desired to compensate for camera motion. Note that a related problem is video stabilization, which aims to remove *unwanted* camera motion, while GMC removes both *intentional* and *unwanted* camera motion [5].

Normally, GMC estimates the homography transformation between two consecutive frames by matching keypoints on the frames, and maps the second frame to a global co-

Pages 21.1-21.11
DOI: https://dx.doi.org/10.5244/C.29.21

ordinate. To remedy outliers in keypoint matches, robust techniques are proposed for homography estimation, e.g., RANSAC [7], by assuming the number of outliers to the correct homography is much less than inliers. However, in the presence of *predominant foreground*, i.e., moving objects and people, a larger proportion of the putative matches are mismatches.

Predominant foreground may result from a higher percentage of coverage by foreground pixels, or occlusion, textureless and non-informative background, blurred background (e.g., camera following the foreground motion), or a combination of these reasons. In presence of predominant foreground, the common variations of RANSAC have little chance of selecting a minimal set of background keypoints by random sub-sampling in a limited number of iterations. Despite its importance, the predominant foreground problem has been overlooked in both video stabilization and GMC algorithms. Even for algorithms designed explicitly for robustness to foreground motion [5, 6, 12], predominant foreground is reported to cause failure. Since GMC estimates homography between consecutive frames and then uses a cascade of homographies to map the current frame to the global motion-compensated coordinate, failure in GMC at a single frame affects all the subsequent frames. This renders the predominant foreground problem very common and significant. Thus, GMC robustness is highly desirable. GMC problem is also aggravated as speed of foreground motion increases, e.g., in sports videos. We qualitatively investigate 500 videos from Sports Videos in Wild (SVW) dataset [16], and observe 35% failure, i.e., background instability, by the baseline method of MLESAC [20], in contrast to 5.1% failure for the proposed method. This demonstrates that the robustness problem is very common and severe for real-world videos.

The main contribution of this paper is a robust GMC (RGMC) method for suppressing foreground keypoint matches and mismatches, enabling a reliable homography estimation in presence of predominant foreground and textureless background. Also, we propose a novel and efficient probabilistic model for homography verification that considers keypoint matching error and consistency of the image edges after warping, and benefits from motion history gleaned from prior matched frames. We demonstrate the superiority of RGMC on challenging videos from three video datasets, when compared with state-of-the-art methods.

1.1 Previous Work

Due to existence of outliers, robust techniques are widely used for homography estimation, e.g., RANSAC [7] and its variants such as Locally-Optimized RANSAC [4], MLESAC [20] and Guided-MLESAC [19]. While RANSAC aims to maximize the number of inliers, MLESAC searches the best hypothesis that maximizes the likelihood via RANSAC, assuming that the inliers are Gaussian distributed and outliers are distributed randomly. To handle the same outlier issue, [10] directly rejects unreliable keypoint matches. However, in case of predominant foreground, problematic matches from the foreground are not unreliable in terms of appearance. Recent works focus on estimating the best or multiple homographies in case of multi-plane background [1, 13, 18, 21, 27]. For instance, Uemura et al. [21] segment each frame to multiple regions denoting different *planes* in the background and find the dominant plane for homography estimation. In contrast, we segment the frame to *foreground* and *background* regions by analyzing motion vector clusters, and remove foreground for robust GMC.

Yan et al. [26] propose a probabilistic framework to combine keypoint matching and appearance similarity to enhance estimation robustness. To model the latter, correlation coefficient between pixels is used. Despite the improved estimation accuracy, for textureless

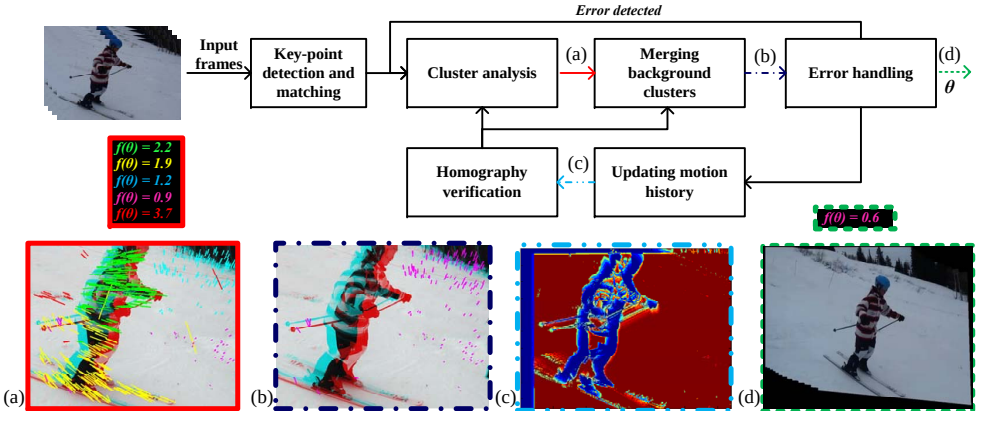

Figure 1: RGMC algorithm flowchart: (a) color indicates various motion vector clusters, (b) the merged cluster of background, (c) the motion history, and (d) the motion compensated video.

background the performance deteriorates. For large foreground, [26] tends to remove foreground, instead of background, motion. In contrast, we use edge matching as an appearance similarity measure with a higher sensitivity and lower computational costs. Motion history-based foreground suppression minimizes its interference with homography estimation.

If camera motion is modeled as 2D translation, simpler methods can be used for GMC. In [3], video stabilization is conducted using the cross-correlation between horizontal and vertical projection of the consecutive frames, by assuming that the largest variation between frames is due to 2D translation. [5] uses the same idea to estimate 2D translation. To improve the robustness to moving foreground, a RANSAC-like approach on projections of bands of the image is utilized. However, [5] fails if the foreground object is too large or the background is textureless, and the simplistic model of 2D translation is easily violated in real-world videos. Thus, we design our RGMC algorithm to minimize the effect of textureless background and large foreground on homography estimation.

2 Proposed Method

The main objective of Robust Global Motion Compensation (RGMC) algorithm is to be robust to the presence of predominant foreground. Thus, it is critical to suppress the foreground and rely on keypoint matches of the background for global motion estimation. We perform *foreground suppression* by clustering motion vectors computed from keypoint matches and identifying potential clusters corresponding to the background, which are merged to provide a set of background keypoints for final homography estimation. As a key enabler for RGMC, a novel and reliable *homography verification model* is presented to consider keypoint matching error and consistency of the edges of images after transformation, and benefit from motion history gleaned from previous frames. Fig. 1 shows the flowchart of the RGMC algorithm, with details presented in the following two subsections.

2.1 Foreground Suppression

We use SURF [2] algorithm for keypoint detection and description. To detect sufficient background keypoints, the Fast-Hessian keypoint detection threshold, τ_s, is decreased drastically. This helps in the cases of nearly uniform and textureless background, or blurred background

Algorithm 1: Robust Global Motion Compensation

Data: Frames $\mathbf{I}_t$ and $\mathbf{I}_{t-1}$ and keypoints matches $\mathbf{D}$, prior homography θ_{t-1} and CFV $f(\theta_{t-1})$ and $f(\theta_{t-2})$
Result: Estimated homography θ_t and motion history $\mathbf{M}_t$

```
Compute the set of motion vectors V from D;
repeat
    Cluster D into D_i (i ∈ {1,..,K}) based on V, set f_i = ∞;
    for i=1 to K do
        while Number of iterations < T_C do
            Randomly select four matching keypoints Q from D_i ;
            if H(Q) > p_{H,0.9} then
                Find homography θ̂_t;
                if At least λ% of keypoints in D_i are inliers for θ̂_t then
                    Calculate the cost function f̂ via Eqn. 10;
                    f_i ← min(f̂, f_i).
    Regularize D_i to D̄_i by randomly selecting a maximum of C matches for each cluster;
    Sort the f_i's in an ascending order and find the sorting index j(i), set
    m_i = ∞, (i ∈ {0,..,K}), i = 0;
    repeat
        i ← i+1 and merge the top i clusters: M_i = ⋃_{k=j(1)}^{j(i)} D̄_k;
        while Number of iterations < T_M do
            Randomly select four matching keypoints Q from M_i ;
            if H(Q) > p_{H,0.9} then
                Find homography θ̂ and calculate the cost function f̂ via Eqn. 10;
                if f̂ < m_i, then θ_i ← θ̂ and m_i ← f̂.
    until m_i > m_{i-1} ∧ i < K;
    θ_t = θ_{i-1}, f(θ_t) = m_{i-1};
until f(θ_t) < η(f(θ_{t-1}) + f(θ_{t-2}))/2 ∨ Number of iterations < T_E;
Update motion history via Eqn. 6 and output θ_t and f(θ_t).
```

due to rapid camera motion (e.g., videos shot by smartphones). However, this also implies that more keypoints will reside on the foreground, which calls for an effective foreground suppression.

Cluster analysis For foreground suppression, the motion vectors resulting from keypoint matches between consecutive frames are clustered. Since motion vectors on the background result from camera motion and are more *consistent* than foreground motion vectors, clustering will likely lead to some candidate regions from the background (see Fig. 1 (a)). Each cluster is analyzed separately by random subsampling of matches in that cluster and evaluating the resultant homography against the cost function, discussed in Sec. 2.2.

Merging background clusters Due to the zooming or motion corresponding to different planes of the background, and not knowing the optimal number of clusters a priori, we allow an over-clustering of K clusters. Thus, background motion vectors may be assigned to multiple clusters. To merge background clusters, based on the estimated homography and cost function value (CFV) of each cluster, a subset of the best clusters are selected to be merged in a greedy algorithm (Fig. 1(b)). Prior to merging, the set of keypoints belonging to each

cluster are regularized by randomly selecting a maximum of C pairs for each cluster. Given that the keypoint matches in background cluster are similar, the regularization has negligible impact on the RGMC accuracy, but remedies the case when part of the foreground (generally with a higher number of matches) is mistakenly merged to the background clusters.

Error handling For GMC applications such as video stitching or pre-processing for motion analysis, failed compensation and homography estimation for a single frame deteriorates the overall performance drastically. Since the context in consecutive frames are similar, we utilize the historical values of the cost function to assist the error handling. If the minimum CFV of homography estimation at the current frame pair is significantly higher than those of previous pairs, we repeat the estimation process with the hope that the randomness in the algorithm will recover the error.

Note that the significance of foreground suppression would be more obvious when plenty of keypoints belong to the foreground, while a few belong to the background. For instance, if foreground has 200 keypoints and background has 10, a RANSAC-like algorithm needs to run 450,000 iterations to ensure a 90% probability of selecting a quadruplet of keypoints from background. However, by analyzing each cluster separately, RGMC efficiently focuses on background matches. Algorithm 1 summarizes the proposed RGMC algorithm. Details of the homography verification model used in the algorithm will be presented next.

2.2 Homography Verification Model

To evaluate the estimated homography from a quadruplet of keypoints matches, we derive a cost function that unifies the keypoint matching score, edge matching score, and the information from compensating previous frames. Denote the matching frames as $\mathbf{I}_{t-1}$ and $\mathbf{I}_t$, their candidate homography as θ_t, and the set of keypoint matches under study as $\mathbf{D}$. In Bayesian framework, similar to [26], θ_t can be estimated by maximizing

$$p(\theta_t|\mathbf{D},\mathbf{I}_t,\mathbf{I}_{t-1},\theta_{t-1}) = \frac{p(\mathbf{D},\mathbf{I}_t,\mathbf{I}_{t-1}|\theta_t,\theta_{t-1})p(\theta_t|\theta_{t-1})}{p(\mathbf{D},\mathbf{I}_t,\mathbf{I}_{t-1}|\theta_{t-1})}, \tag{1}$$

where θ_{t-1} is the obtained prior homography of frames $\mathbf{I}_{t-1}$ and $\mathbf{I}_{t-2}$. The $p(\theta_t|\theta_{t-1})$ is the conditional probability of θ_t given the prior homography θ_{t-1}. The denominator of Eqn. 1 is constant w.r.t. θ_t. By expanding the likelihood term, the homography can be verified using

$$p(\theta_t|\mathbf{D},\mathbf{I}_t,\mathbf{I}_{t-1},\theta_{t-1}) \propto p(\mathbf{D}|\mathbf{I}_t,\mathbf{I}_{t-1},\theta_t,\theta_{t-1})p(\mathbf{I}_t,\mathbf{I}_{t-1}|\theta_t,\theta_{t-1})p(\theta_t|\theta_{t-1}). \tag{2}$$

The term $p(\mathbf{D}|\mathbf{I}_t,\mathbf{I}_{t-1},\theta_t,\theta_{t-1}) = p(\mathbf{D}|\mathbf{I}_t,\mathbf{I}_{t-1},\theta_t)$ and represents how well the keypoint matches $\mathbf{D}$ extracted from $\mathbf{I}_t$ and $\mathbf{I}_{t-1}$ are matched by θ_t. Knowing $\mathbf{I}_t$ is independent from θ_{t-1}, the term $p(\mathbf{I}_t,\mathbf{I}_{t-1}|\theta_t,\theta_{t-1}) = p(\mathbf{I}_t,\mathbf{I}_{t-1}|\theta_t)$, and reflects how well the frame $\mathbf{I}_t$ transformed under θ_t, denoted as $\mathbf{I}_{t|\theta_t}$, matches $\mathbf{I}_{t-1}$. Thus, the homography is estimated by minimizing,

$$\theta_t^* = \arg\min_{\theta_t}[-\ln(p(\mathbf{D}|\mathbf{I}_t,\mathbf{I}_{t-1},\theta_t)) - \ln(p(\mathbf{I}_t,\mathbf{I}_{t-1}|\theta_t)) - \ln(p(\theta_t|\theta_{t-1}))]. \tag{3}$$

Keypoint matching error Based on the analysis of Yan et al. [26], the keypoint matching error for inliers, $p(\mathbf{D}_{\mathbf{in}}|\mathbf{I}_t,\mathbf{I}_{t-1},\theta_t)$, is better represented by a Laplacian model than the conventional Gaussian model. Denote (x_R^i, y_R^i) and (x_T^i, y_T^i) as the ith matching keypoint coordinates of $\mathbf{I}_t$ and $\mathbf{I}_{t-1}$ respectively, transformation of (x_R^i, y_R^i) under θ_t as (x_{RT}^i, y_{RT}^i), transformation of (x_T^i, y_T^i) under θ_t^{-1} as (x_{TR}^i, y_{TR}^i) , and $d_i \leftarrow |x_{TR}^i - x_R^i| + |y_{TR}^i - y_R^i| + |x_{RT}^i - x_T^i| + |y_{RT}^i - y_T^i|$. We use the same method as [26] to compute the keypoint matching error,

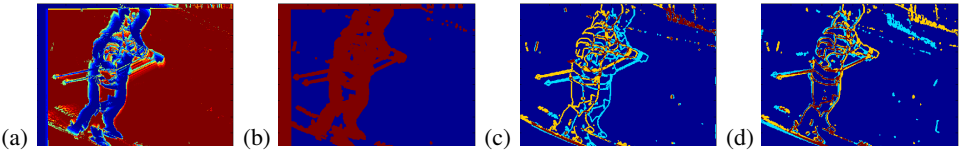

Figure 2: (a) Motion history $\mathbf{M}_t$, (b) Mask $\mathbf{M} = \mathbb{I}(\mathbf{M}_t > \tau)$, (c) edge matching for an accurate θ_t that matches the background, (d) edge matching for an inaccurate θ_t that matches the foreground.

$$p(\mathbf{D}_{in}|\mathbf{I}_t, \mathbf{I}_{t-1}, \theta_t) = \prod_{i=1}^{N_{in}} \frac{1}{16b^4} e^{-\frac{d_i}{b}} \tag{4}$$

where N_{in} is the number of inliers and the scale b is the Laplacian distribution parameter. Denoting γ_i as an indicator variable for inlier/outlier and considering that an outlier has a uniform distribution over the entire area of the frame, which is denoted as S, we have

$$p(\mathbf{D}|\mathbf{I}_t, \mathbf{I}_{t-1}, \theta_t) = \prod_{i=1}^{|\mathbf{D}|} [\gamma_i \frac{1}{16b^4} e^{-\frac{d_i}{b}} + (1-\gamma_i)\frac{1}{S^2}]. \tag{5}$$

Appearance consistency The appearance consistency under θ_t transformation, $p(\mathbf{I}_t, \mathbf{I}_{t-1}|\theta_t)$, is normally computed via pixel-based correlation [26]. We propose edge-based matching for multiple reasons. First, the pixel-based matching score is not sensitive enough for textureless background, e.g., a homography with error of few pixels displacement leads to similar scores as a perfect match. In contrast, the tolerance for error is much lower by matching the edges, which results in more accurate homography models. Although low-texture images produce few and generally noisy edge pixels, our experiments show that edge matching outperforms pixel-based correlation, even in low-texture conditions, similar to the results reported in [25]. Second, when stitching video frames based on global motion compensation, errors typically occur in mis-matched edges at the boundary of the frames. These errors are very distracting for viewers' visual perception, and they are more likely to be remedied by edge-based appearance matching. Finally, in pixel matching, time-consuming image warping is needed for computing $\mathbf{I}_{t|\theta_t}$. Edge matching only needs to warp edge pixels in $\mathbf{I}_t$, leading to a typical $10\times$ speed-up over pixel matching.

To assure that the edge matching score reflects how well the background, not foreground, of the two frames match, we iteratively update a motion history $\mathbf{M}_t$ (see Fig. 1 (c)) as,

$$\mathbf{M}_t \leftarrow \alpha \mathbf{M}_{t-1} + (1-\alpha)|\mathbf{I}_{t-1} - \mathbf{I}_{t|\theta_t}|, \tag{6}$$

where α is a weighting scalar within 0 and 1, and $|.|$ denotes the element-wise absolute value operator. We define the edge matching score (EMS) as,

$$E(\mathbf{I}_1, \mathbf{I}_2, \mathbf{R}) = \frac{2\|\Phi(\mathbf{I}_1) \bigodot \Phi(\mathbf{I}_2) \bigodot \mathbf{R}\|_1}{\|\Phi(\mathbf{I}_1) \bigodot \mathbf{R}\|_1 + \|\Phi(\mathbf{I}_2) \bigodot \mathbf{R}\|_1 + c}, \tag{7}$$

where Φ is edge detection operator, $\bigodot$ is element-wise multiplication, $\mathbf{R}$ denotes the mask specifying the region of interest for EMS calculation, $\|\cdot\|_1$ computes the L_1 matrix norm, and $c(=0.001)$ is a constant to avoid division by zero. $E(\mathbf{I}_1, \mathbf{I}_2, \mathbf{R})$ ranges between 0 and 1 with 1 representing a perfect match. In Eqn. 3, we use $E(\mathbf{I}_{t-1}, \mathbf{I}_{t|\theta_t}, \mathbf{M})$, where $\mathbf{M} = \mathbb{I}(\mathbf{M}_t > \tau)$ is obtained by thresholding the motion history and $\mathbb{I}(\cdot)$ is an indicator function. Fig. 2 shows a motion history and edge matching results for two candidate θ_t's. We will later discuss how the probability model for E is obtained.

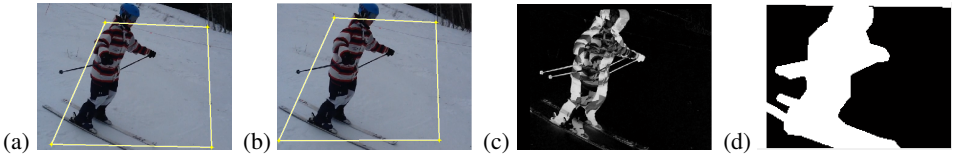

Figure 3: (a-b) Two consecutive frames and the matched quadruplet by the labeler, (c) the absolute difference of two frames matched via the quadruplet in (a,b), (d) manually labeled foreground mask.

Conditional homography distribution Based on our experiments, and also prior work [5] on YouTube Action Dataset [11], the largest variation between consecutive video frames is due to 2D translation. Thus, to utilize the prior information of θ_{t-1} for a stable homography estimation, we decompose the homography model into translation, scale, and rotation models [22]. Denote the absolute difference in components of θ_t and θ_{t-1} after decomposition as t_x and t_y for translation, Δs for scale and $\Delta\alpha$ for rotation angle. Assuming independence among components, we define

$$p(\theta_t|\theta_{t-1}) = p(t_x)p(t_y)p(\Delta s)p(\Delta\alpha). \tag{8}$$

Quadruplet filtering RGMC evaluates a large number of quadruplets of keypoint matches, and computes their EMS. To improve the efficiency, we filter the candidate quadruplets before the optimization of Eqn. 3. Intuitively, if the keypoint in the quadruplet are spatially close to each other, it is less likely to have an accurate estimate of θ_t, because homography estimation is more sensitive to the accuracy of keypoint locations. Also, background keypoints have generally a higher spatial dispersion than the foreground keypoints. Thus, only if the entropy (or dispersion) of a candidate quadruplet is above a threshold, we fully evaluate the cost function. Specifically, we use m-spacing estimate of entropy [9], similar to [5], as

$$H = \frac{1}{n}\sum_{i=1}^{n-m} \ln\left(\frac{n}{m}(x_{i+m} - x_i)\right), \tag{9}$$

where m is the spacing parameter (set to 1) and n is number of points. We first sort the x values prior to using them in Eqn. 9. Entropy estimates of x and y coordinates of the quadruplet are calculated separately and the minimum of them is the entropy of the quadruplet.

Model training Having presented the Bayesian framework, we now introduce our empirical approach to learn the various probability models. For this learning, we manually stitch 250 pairs of consecutive frames to find the best homography estimate. The labeler uses our developed GUI to match four background keypoints in two consecutive frames and fine tune the matches to visually minimize the background stitching error. The labeler also specifies a foreground mask, representing the region resulted from foreground movement. Fig. 3 shows two consecutive video frames and the manually matched quadruplets. From the manually labeled sequences, we find the empirical distribution of E, t_x, t_y, Δs, $\Delta\alpha$, and H. As shown in Fig. 4, E, Δs, $\Delta\alpha$, and H are well approximated by a normal distribution. For H distribution, 10% percentile ($p_{H,0.9}$), reflecting the value that 90% of observed point entropies are larger than, is also shown. For t_x and t_y, Laplacian distribution is more appropriate. By plugging the probability models to Eqn. 3 and ignoring the constants, the final cost function is,

$$f(\theta_t) = \frac{\sum_{i=1}^{N_{in}} \frac{d_i}{b} + \sum_{i=1}^{N_{out}} \ln(S^2)}{N_{in}+N_{out}} + \frac{(E(\mathbf{I}_t,\mathbf{I}_{t-1},\mathbf{M}) - \mu_E)^2}{2\sigma_E^2} + \lfloor\frac{(\Delta s - \mu_{\Delta s})^2}{2\sigma_{\Delta s}^2}\rfloor_T + \lfloor\frac{(\Delta\alpha - \mu_{\Delta\alpha})^2}{2\sigma_{\Delta\alpha}^2}\rfloor_T + \lfloor\frac{|\Delta t_x - \mu_{\Delta t_x}|}{b_{t_x}}\rfloor_T + \lfloor\frac{|\Delta t_y - \mu_{\Delta t_y}|}{b_{t_y}}\rfloor_T, \tag{10}$$

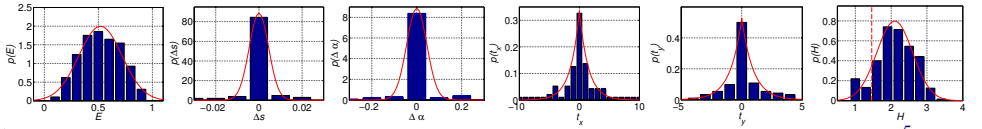

Figure 4: Empirical and fitted distributions for (a) $E \sim N(0.52, 0.04)$, (b) $\Delta s \sim N(0, 2 \times 10^{-5})$, (c) $\Delta\alpha \sim N(0, 2 \times 10^{-3})$, (d) $t_x \sim Laplace(0, 1.50)$, (e) $t_y \sim Laplace(0, 0.95)$, and (f) $H \sim N(2.1, 0.25)$.

Figure 5: Sample frames of the test videos in (a) SVW, (b) HMDB51, and (c) Holleywood2 datasets.

where N_{out} is number of outliers and $\lfloor x \rfloor_T = \min(x, T)$ restricts the impact of prior information. Since keypoint matching error is dependent on the number of keypoints, we normalize it with the total number of keypoints. The homography θ_t is estimated by

$$\theta_t^* = \arg\min(f(\theta_t)). \tag{11}$$

3 Experimental Results

This section presents the experimental results of RGMC, and its comparison with our implementations of the RANSAC variation called MLESAC [20] and the HEASK method [26].

Dataset We select 50 videos from SVW dataset [16], where 24 videos are used for model learning in Sec. 2.2, and the rest for testing. SVW contains videos of amateurs practicing a sport, shot using smartphone by ordinary people. Thus, highly unconstrained SVW is an excellent example of user-generated videos with predominant foreground of humans. We also use 10 videos from Holleywood2 [14] and 15 videos from HMDB51 [8] datasets[1]. In total, 51 videos are used for quantitative evaluation with sample frames shown in Fig. 5.

Parameters In all the experiments, we have the same fixed parameter setting, i.e., $\tau = 0.5$, $C = 50$, $T_C = 50$, $T_M = 100$, $T_E = 2$, $K = 10$, $\eta = 1.5$, $\alpha = 0.5$, $\lambda = 70\%$, and $T = 100$. Our experiments show that RGMC is robust to variation of parameters. The most important parameter is K. Large values of K increase the computational cost. On the other hand, K should be large enough so that foreground, background, and erroneous matches are mapped to different clusters. As a trade-off, we use $K = 10$.

[1]For these two datasets, videos are temporally trimmed around the signature motion in the video, practically disabling effect of our motion history module. In HMDB51, similar to many existing datasets such as UCF101 [17], the video resolution is only 320×240, thus GMC suffers from both video content and the low resolution.

Algorithm	Ground Truth	MLESAC		HEASK		RGMC				
Setting	–	DT	LT	DT	LT	(20,50)	(**50**,**100**)	(100,200)	D-M	D-E
BRE ($\times 10^{-3}$)	**7.59**	**15.65**	18.59	17.33	**14.24**	11.77	10.11	**10.02**	11.60	11.25

Table 1: Impact of different settings on average BRE for each algorithm. DT and LT denote default ($\tau_s = 1000$) and lowered ($\tau_s = 100$) detection threshold in SURF algorithm, respectively. For RGMC $\tau_s = 100$ is used and 3 different setting of (T_C, T_M) are reported. D-M and D-E denote default setting of $(T_C, T_M) = (50, 100)$ with motion history and error handling turned off, respectively.

Evaluation metric For accuracy evaluation, we have manually matched a quadruplet of keypoints and found the ground truth homography θ_0 for a total of 350 pairs of consecutive frames in challenging periods in 51 test videos. The same GUI described in Sec. 2.2 is used to obtain θ_0 and the foreground mask. We denote the intersection of the complement of this mask, i.e., the background mask, and the region covered by $\mathbf{I}_{t|\theta_0}$, as $\mathbf{B}$. We quantify the consistency of frames $\mathbf{I}_t$ and $\mathbf{I}_{t-1|\theta}$ (grayscale frames with pixels ranging between 0 and 1) using the background region error (BRE), $\varepsilon = \frac{1}{\|\mathbf{B}\|_1} \| |(\mathbf{I}_{t-1} - \mathbf{I}_{t|\theta_t})| \bigodot \mathbf{B} \|_1$.

Accuracy assessment Table 1 represents the average BRE on test videos for different algorithms. Due to random nature of algorithms, we repeat each experiment 5 times and report the average performance. To ensure that comparisons are fair, we decrease the keypoint detection thresholds also for baseline methods. HEASK has better performance with lowered threshold and thus we use this setting for the experiments. We also report results for different iteration numbers T_C and T_M for RGMC and as a trade-off between accuracy and efficiency, select $(T_C, T_M) = (50, 100)$ as default values for RGMC. In addition, we turn off the modules of *Motion History* and *Error Handling* in RGMC alternatively, to verify that their existence is helpful. Fig. 6 shows two consecutive frames of three sample videos matched by different algorithms, along with the ground truth matching. As shown, RGMC produces very accurate background matching. Fig. 7 represents the average per-video BRE, sorted by the BRE of ground truth matching. As shown, RGMC performance is very robust and in most videos RGMC matching error is very close to the ground truth value. Finally, Fig. 8 compares stitching results on a sample video using different algorithms. It is worth noting that since a cascade of homographies are used for GMC and stitching of video frames, propagation of errors of matching consecutive frames, gives rise to inaccuracy as the length of the input video increases. Also, coexistence of textureless background and large foreground (in terms of the total number of pixels covered by the foreground), may cause failure in the RGMC algorithm, especially if the foreground motion exists starting the initial frames.

Computational cost For the comparison with baseline methods, we test Matlab implementations of algorithms on a PC with Intel i5-3470@2GHz CPU. The average time for matching frame pair of size $720 \times 1,280$ (480×854) by MLESAC, HEASK, and RGMC is 2.0 (0.3), 53.1 (21.3), and 4.3 (2.3) seconds, respectively. We also have a C++ implementation of RGMC using the OpenCV libraries, which takes 1.4 (0.7) seconds for matching a frame pair of size $720 \times 1,280$ (480×854)[2].

Qualitative evaluation In addition to the aforementioned quantitative study, we also perform the qualitative evaluation on *unlabeled* videos to demonstrate the severity of the predominant foreground issue in real-world videos, and the superiority of RGMC on a large scale dataset. For each video, we run a GMC algorithm, visually observe the motion-compensated videos, and claim a *failure* if an instable background is observed (e.g., see

[2]Source code is available at http://www.cse.msu.edu/∼liuxm/RGMC

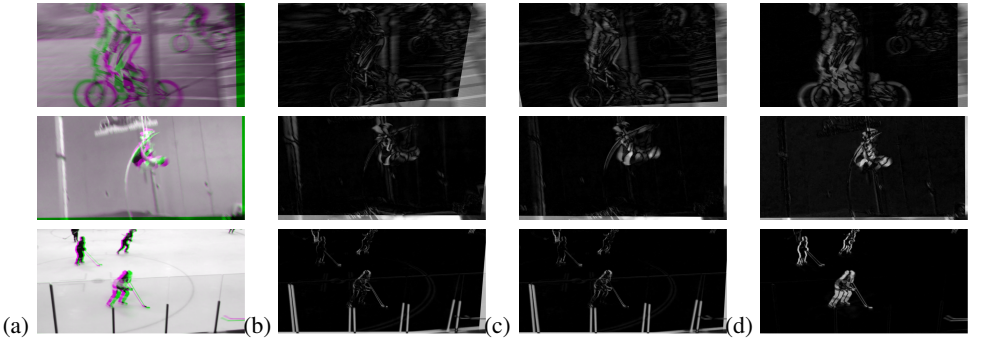

Figure 6: Each row shows GMC results of two consecutive frames from video ID S17, S19, and S9 by (a) manual labeling, (b) MLESAC, (c) HEASK, and (d) RGMC. In (a), colorful pixels show the pixels that are different between overlaid frames. In (b-d), the pixel brightness indicates the difference.

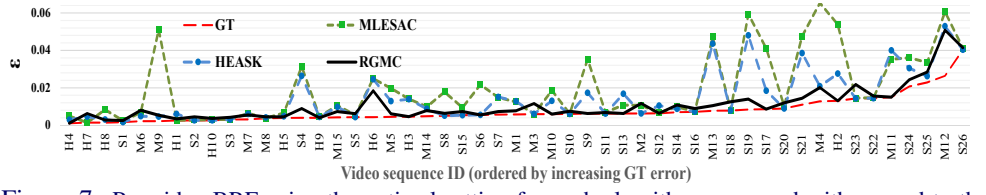

Figure 7: Per-video BRE using the optimal setting for each algorithm compared with ground truth (GT) matching BRE. Video ID is according to Fig. 5 and 6.

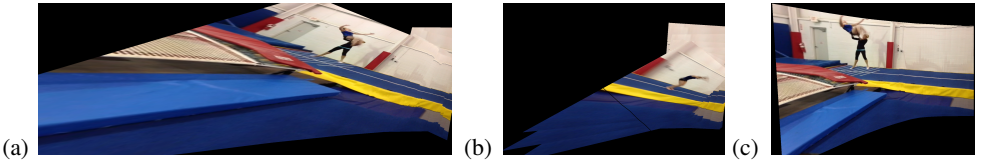

Figure 8: A 40-frame sequence of gymnastics backflips in textureless background stitched using (a) MLESAC, (b) HEASK, and (c) RGMC. Consistency of the background shows the superiority of RGMC. For HEASK, stitching up to frame #10 is shown, after which the stitching drastically fails.

Fig. 8 (a,b)). We observe a failure rate of 32% by the MLESAC method among 225 videos from three categories of cartwheel, dive and dribble in HMDB51 dataset. Further, a 35% failure rate by MLESAC is observed from 500 videos of SVW dataset; in contrast on the same data our RGMC has merely a 5% failure rate.

4 Conclusions

We presented a robust global motion compensation (RGMC) algorithm that delivers reliable results in the presence of predominant foreground and textureless or blurry background, enabling its application to real-world unconstrained videos. By foreground suppression, RGMC is able to tolerate large foreground and occlusion. Also, the proposed method successfully handles keypoint matching with a very low matching threshold, required for GMC in low texture background. This is achieved by clustering motion vectors, and analyzing each cluster to identify matches pertaining to the background. A novel homography verification model is proposed to support the RGMC. Extensive experiments and comparison with manually matched ground truth and baseline methods demonstrate the superiority of RGMC.

References

[1] Adam Barclay and Hannes Kaufmann. FT-RANSAC: Towards robust multi-modal homography estimation. In *Proc. IAPR Workshop PRRS*, pages 1–4. IEEE, 2014.

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded up robust features. In *Proc. ECCV*, pages 404–417. Springer, 2006.

[3] Angelo Bosco, Arcangelo Bruna, Sebastiano Battiato, Giuseppe Bella, and Giovanni Puglisi. Digital video stabilization through curve warping techniques. *IEEE Trans. Consumer Electronics*, 54(2):220–224, 2008.

[4] Ondřej Chum, Jiří Matas, and Josef Kittler. Locally optimized RANSAC. *Pattern Recognition*, pages 236–243, 2003.

[5] Oscar Déniz, Gloria Bueno, E Bermejo, and Rahul Sukthankar. Fast and accurate global motion compensation. *Pattern Recognition*, 44(12):2887–2901, 2011.

[6] Rahul Dutta, Bruce Draper, and J Ross Beveridge. Video alignment to a common reference. In *IEEE Winter Conf. WACV*, pages 808–815. IEEE, 2014.

[7] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *ACM Trans. Communications*, 24(6):381–395, 1981.

[8] Hilde Kuehne, Hueihan Jhuang, Rainer Stiefelhagen, and Thomas Serre. HMDB51: A large video database for human motion recognition. In *High Performance Comput. Sci. Eng.*, pages 571–582. Springer, 2013.

[9] Erik G Learned-Miller et al. ICA using spacings estimates of entropy. *J. Machine Learning Research*, 4: 1271–1295, 2003.

[10] Xiangru Li and Zhanyi Hu. Rejecting mismatches by correspondence function. *Int. J. Comput. Vision*, 89(1): 1–17, 2010.

[11] Jingen Liu, Jiebo Luo, and Mubarak Shah. Recognizing realistic actions from videos in the wild. In *Proc. IEEE Conf. CVPR*, pages 1996–2003. IEEE, 2009.

[12] Shuaicheng Liu, Lu Yuan, Ping Tan, and Jian Sun. Steadyflow: Spatially smooth optical flow for video stabilization. In *Proc. IEEE Conf. CVPR*, pages 4209–4216. IEEE, 2014.

[13] Jiayi Ma, Jun Chen, Delie Ming, and Jinwen Tian. A mixture model for robust point matching under multi-layer motion. *PloS one*, 9(3):e92282, 2014.

[14] Marcin Marszałek, Ivan Laptev, and Cordelia Schmid. Actions in context. In *Proc. IEEE Conf. CVPR*, 2009.

[15] S. Morteza Safdarnejad, Xiaoming Liu, and Lalita Udpa. Genre categorization of amateur sports videos in the wild. In *Proc. Int. Conf. ICIP*. IEEE, 2014.

[16] S. Morteza Safdarnejad, Xiaoming Liu, Lalita Udpa, Brooks Andrus, John Wood, and Dean Craven. Sports videos in the wild (SVW): A video dataset for sports analysis. In *Proc. Int. Conf. FG*. IEEE, 2015.

[17] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[18] Roberto Toldo and Andrea Fusiello. Robust multiple structures estimation with j-linkage. In *Proc. European Conf. ECCV*, pages 537–547. Springer, 2008.

[19] Ben J Tordoff and David W Murray. Guided-mlesac: Faster image transform estimation by using matching priors. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 27(10):1523–1535, 2005.

[20] Philip HS Torr and Andrew Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Comput. Vision and Image Understanding*, 78(1):138–156, 2000.

[21] Hirofumi Uemura, Seiji Ishikawa, and Krystian Mikolajczyk. Feature tracking and motion compensation for action recognition. In *Proc. BMVC*, pages 1–10, 2008.

[22] M Vargas and E Malis. Deeper understanding of the homography decomposition for vision-base control. *Unité de recherche INRIA Sophia Antipolis (France)*, 2007.

[23] Heng Wang, Alexander Klaser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *Proc. IEEE Conf. CVPR*, pages 3169–3176. IEEE, 2011.

[24] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *Int. J. Comput. Vision*, 103(1):60–79, 2013.

[25] Lu Wang, Ulrich Neumann, and Suya You. Wide-baseline image matching using line signatures. In *Proc. Int. Conf. ICCV*, pages 1311–1318. IEEE, 2009.

[26] Qing Yan, Yi Xu, Xiaokang Yang, and Truong Nguyen. HEASK: Robust homography estimation based on appearance similarity and keypoint correspondences. *Pattern Recognition*, 47(1):368–387, 2014.

[27] Marco Zuliani, Charles S Kenney, and BS Manjunath. The multiransac algorithm and its application to detect planar homographies. In *Proc. Int. Conf. ICIP*, volume 3, pages III–153. IEEE, 2005.