Towards a Unified Approach to Homography Estimation Using Image Features and Pixel Intensities

Lucas Nogueira, Ely C. de Paiva

School of Mechanical Engineering
University of Campinas
Campinas, SP, Brazil
[lucas.nogueira]|[ely]@fem.unicamp.br

Geraldo Silveira

Robotics and Computer Vision research group
Center for Information Technology Renato Archer
Campinas, SP, Brazil
Geraldo.Silveira@cti.gov.br

***Abstract*—The homography matrix is a key component in various vision-based robotic tasks. Traditionally, homography estimation algorithms are classified into feature- or intensity-based. The main advantages of the latter are their versatility, accuracy, and robustness to arbitrary illumination changes. On the other hand, they have a smaller domain of convergence than the feature-based solutions. Their combination is hence promising, but existing techniques only apply them sequentially. This paper proposes a new hybrid method that unifies both classes into a single nonlinear optimization procedure, applies the same minimization method, and uses the same homography parametrization and warping function. Experimental validation using a classical testing framework shows that the proposed unified approach has improved convergence properties compared to each individual class. These are also demonstrated in a visual tracking application. As a final contribution, our ready-to-use implementation of the algorithm is made publicly available to the research community.**

Keywords–Robot vision; Homography optimization; Hybrid approaches; Vision-based applications.

I. Introduction

The homography matrix is a key component in computer vision. It relates corresponding pixel coordinates of a planar object in different images, and has been used in a variety of vision-based applications such as image mosaicing [1], visual servoing [2] and object grasping [3]. The homography estimation task can be formulated as an Image Registration (IR) problem. IR can be defined as a search for the parameters that best define the transformation between corresponding pixels in a pair of images. Solutions to this problem involve the definition of at least four important characteristics [4]: the information space, the transformations models, the similarity measures, and the search strategy.

With respect to the information space, the vast majority of vision-based algorithms use a Feature-Based (FB) approach. In this class, firstly an extraction algorithm searches each image for geometric primitives and selects the best candidates. Then, a matching algorithm establishes correspondences between features in different images. Afterwards, the actual estimation takes place. However, both the extraction and matching steps are error-prone and can produce outliers that affect the quality of the estimation. Additionally, by using only a sparse set of features, these algorithms may discard useful information.

In contrast, Intensity-Based (IB) methods have no extraction and matching steps. These methods are also referred to as direct methods since they exploit the pixel intensity values directly. This allows the estimation algorithm to work with more information than FB methods and does not depend on particular primitives. Thus, it leads to more accurate estimates and is highly versatile. However, an important drawback is that they require a small interframe displacement, i.e., a sufficient overlapping between consecutive images.

The algorithms presented in this work use multidimensional optimization methods as the main search strategy for the image registration problem. When formulated as such, an initial solution is iteratively refined using a nonlinear optimization method. Specifically, the algorithms presented here are derived from the Efficient Second-order Minimization method (ESM) [5]. Its advantages include both a higher convergence rate and a larger convergence domain than standard iterative methods. It allows for a second-order approximation of the Taylor series without computationally expensive Hessian calculations.

The use of the ESM framework has shown remarkable results for IB methods. However, its application within FB methods has been limited so far. As discussed, the two classes of estimation methods have complementary strengths. This work aims to develop a hybrid method that exploits their advantages and reduces their shortcomings. The proposed algorithm is made available as ready-to-use ROS [6] packages and as a C++ library. In particular, a homography-based visual tracking application is also developed. In summary, our contribution is the development of a vision-based algorithm that:

- unifies the intensity- and feature-based approaches to homography estimation into a single nonlinear optimization problem;
- solves that problem using the same efficient minimization method, homography parametrization, and warping function;
- can be applied in real-time settings, such as for homography-based visual tracking as experimentally demonstrated in this paper; and
- its ready-to-use implementation is made publicly available for research purposes as a C++ library and as a ROS package.

The remainder of this article is organized as follows. Section II presents the related works, whereas Section III describes the proposed unified approach. Section IV then reports the benchmarking experiments and the application of the proposed algorithm to visual tracking. Finally, the conclusions are drawn in Section V, and some references are given for further details.

 ISBN: 978-1-61208-787-0

II. Related Works

The main distinction between IB and FB methods regards their information space. Indeed, on one hand FB requires the extraction and association of geometric primitives in different images before the actual estimation can occur. These primitives can be, e.g., points and lines [1][7]. IB methods simultaneously solves for the estimation problem and pixel correspondences with no intermediate steps [8][9].

The transformation model dictates which parameters are estimated. For example, the original Lucas-Kanade [10] algorithm only estimated translations in the image space. This was later extended to more sophiscated warp functions [11]. Simultaneous Localization And Mapping (SLAM) algorithms commonly use IR to perform the pose and structure estimation [12]. The homography matrix is often used as a transformation model when dealing with predominantly planar regions of interest [13][14][15]. Illumination parameters may also be considered as a component of the transformation model, e.g., in [16].

The quality of the IR is defined by a similarity measure. When an optimization method is applied, this measure is often used as a cost function, such as the Sum of Squared Differences (SSD) [10][17]. Other possibilities include correlation-based metrics [18][19] and mutual information [20].

The last component of IR algorithms is the search strategy. Most real-time applications use a multidimensional optimization approach based on gradient descent. They use the first and second derivatives of the similarity measures with respect to the transformation parameters. The ESM algorithm is such an example, and is applied in the proposed method. Alternative optimization approaches include Gauss-Newton and Levenberg-Marquardt [21]. All of these techniques are most suited to applications with small interframe displacements. Indeed, global techniques are too computationally expensive to be applied in real-time settings. A more thorough review and comparison of image registration algorithms can be found in [22][23].

As for the existing techniques that combine IB and FB methods, their overwhelming majority only applies them sequentially, e.g., [24][25]. In sequential strategies, a FB technique is firstly considered and then its estimated parameters are fed as the initial guess to some IB optimization. This standard combination scheme is thus not optimal and is more time consuming. An exception to that sequential procedure is reported in [26]. However, it aims to estimate the pose parameters, which requires a calibrated camera. The objective of this paper is to estimate the projective homography, i.e., there is no calibrated camera. Furthermore, that existing technique applies a first-order minimization method, and the considered scaling factors do not take into account the convergence properties of the individual approaches, as will be proposed in the sequel.

III. Proposed Unified Approach

Consider that a *reference template* has been specified to an estimation algorithm. This is typically a region of interest with predefined resolution inside a larger reference image. Then, a second image, referred to as the *current image*, is given to that algorithm. The goal is to find the transformation parameters that, when applied to the current image, results in a current template identical to the reference template.

A. Transformation Models

The considered transformation models consist of a geometric and a photometric one. The geometric transformation model explains image changes due to variations in the scene structure and/or the camera motion. For a given pixel $\mathbf{p}^*$ in the reference template that corresponds to pixel $\mathbf{p}$ in the current image, we model the geometric motion using a homography:

$$\mathbf{p} \propto \mathbf{H}\mathbf{p}^* \tag{1}$$

$$= \left[\frac{h_{11}u^* + h_{12}v^* + h_{13}}{h_{31}u^* + h_{32}v^* + h_{33}}, \frac{h_{21}u^* + h_{22}v^* + h_{23}}{h_{31}u^* + h_{32}v^* + h_{33}}, 1 \right]^\top \tag{2}$$

$$= \mathbf{w}(\mathbf{H}, \mathbf{p}^*), \tag{3}$$

where $\mathbf{p}^* = [u^*, v^*, 1]^\top \in \mathbb{P}^2$ is the homogeneous pixel coordinates in the reference template, $\mathbf{w}$ is the warping operator, and $\mathbf{H} \in \mathbb{SL}(3)$ is the projective homography matrix with its elements $\{h_{ij}\}$. Such matrix has only eight degrees-of-freedom. In general, this situation leads to a reprojection step after each iteration of the minimization algorithm that takes the estimated homography into the Special Linear Group. To avoid this problem, the proposed algorithm parameterizes the homography using its corresponding Lie Algebra [2]. This is accomplished via the matrix exponential function, which maps a region around the identity matrix $\mathbf{I} \in \mathbb{SL}(3)$ to a region around the origin $\mathbf{0} \in \mathfrak{sl}(3)$. A matrix $\mathbf{A}(\mathbf{v}) \in \mathfrak{sl}(3)$ is the linear combination of eight matrices that form a base of the Lie Algebra. Therefore $\mathbf{v}$ has eight components. A homography is thus parameterized as

$$\mathbf{H}(\mathbf{v}) = \exp(\mathbf{A}(\mathbf{v})). \tag{4}$$

The homography matrix may be used to extract relative motion and scene structure information [27]. However, this decomposition is out of the scope of this work and is unnecessary for many robotic applications.

The photometric transformation model explains the changes in the image due to variations in the lighting conditions of the scene. Let us model in this work only global illumination variations, i.e., changes that apply equally to all pixels in the images. This model is defined as

$$\mathcal{I}'(\mathbf{p}) = \alpha\,\mathcal{I}(\mathbf{p}) + \beta, \tag{5}$$

where $\mathcal{I}(\mathbf{p}) \geq 0$ is the intensity value of the pixel $\mathbf{p}$, $\mathcal{I}'(\mathbf{p}) \geq 0$ denotes its transformed intensity, and the gain $\alpha \in \mathbb{R}$ and the bias $\beta \in \mathbb{R}$ are the parameters that fully define the transformation. These parameters can be viewed as the adjustments in the image constrast and brightness, respectively.

B. Nonlinear Least Squares Formulation

Consider that the reference template is composed of m pixels. Also, consider that a feature detection and matching algorithm provides n feature correspondences between the reference template and the current image. Ideally, it would be possible to find a vector $\mathbf{x}^* = \{\mathbf{H}^*, \alpha^*, \beta^*\}$ such that:

$$\alpha^*\mathcal{I}(\mathbf{w}(\mathbf{H}^*, \mathbf{p}_i^*)) + \beta^* = \mathcal{I}^*(\mathbf{p}_i^*), \qquad \forall i = 1, 2, \ldots, m, \tag{6}$$

$$\mathbf{w}(\mathbf{H}^*, \mathbf{q}_j^*) = \mathbf{q}_j, \qquad \forall j = 1, 2, \ldots, n, \tag{7}$$

by substituting (3) in (5), where $\mathcal{I}$ and $\mathcal{I}^*$ are the current and reference images, respectively, $\mathbf{p}_i^* \in \mathbb{P}^2$ contains the coordinates of the i-th pixel of the reference template, and $\mathbf{q}_j, \mathbf{q}_j^* \in \mathbb{P}^2$ are the representations of the j-th feature correspondence set in the

 ISBN: 978-1-61208-787-0

current image and reference template, respectively. The perfect calculation of $\mathbf{x}^*$ is impossible due to a variety of reasons, including noise in the camera sensor and outliers in the feature matching. This leads to the reformulation of this task as a nonlinear least-squares problem.

Two separate cost-functions are defined: One for the IB part and another for the FB one. The i-th pixel of the reference template contributes to the following row to the IB cost function via the distance

$$a_i(\mathbf{x}) = \alpha \mathcal{I}(\mathbf{w}(\mathbf{H}, \mathbf{p}_i^*)) + \beta - \mathcal{I}^*(\mathbf{p}_i^*), \tag{8}$$

and an output vector $\mathbf{y}_{IB}$ can be constructed as:

$$\mathbf{y}_{IB} = \begin{bmatrix} a_1 & a_2 & \cdots & a_m \end{bmatrix}^\top. \tag{9}$$

The FB cost function is defined using the distance between the features coordinates in each image:

$$\mathbf{b}_j(\mathbf{x}) = \mathbf{w}(\mathbf{H}, \mathbf{q}_j^*) - \mathbf{q}_j = \begin{bmatrix} b_j^u & b_j^v & 0 \end{bmatrix}, \tag{10}$$

where b_j^u, b_j^v are distances between the features in the u and v directions, respectively. The third element is disregarded since it is always zero. Thus, a vector $\mathbf{y}_{FB}$ can be constructed as:

$$\mathbf{y}_{FB} = \begin{bmatrix} b_1^u & b_1^v & b_2^u & b_2^v & \cdots & b_n^u & b_n^v \end{bmatrix}^\top. \tag{11}$$

Using (9) and (11), a unified nonlinear least squares problem can be defined as

$$\min_{\mathbf{x}=\{\mathbf{H},\alpha,\beta\}} \frac{1}{2}\left(w_{IB} \left\| \mathbf{y}_{IB}(\mathbf{x}) \right\|_2^2 + w_{FB} \left\| \mathbf{y}_{FB}(\mathbf{x}) \right\|_2^2 \right), \tag{12}$$

where w_{IB}, w_{FB} are carefully chosen weights given to the intensity- and feature-based components of the cost function, respectively, as will be proposed later on. For real-time systems, only local optimization methods can be applied since global ones are too costly. In this case, an initial approximation $\widehat{\mathbf{x}} = \{\widehat{\mathbf{H}}, \widehat{\alpha}, \widehat{\beta}\}$ of the true solution is required. This estimate can be integrated into the least-squares formulation as:

$$\min_{\mathbf{z}=\{\mathbf{v},\alpha,\beta\}} \frac{1}{2}\left(w_{IB} \left\| \mathbf{y}_{IB}(\mathbf{x}(\mathbf{z}) \circ \widehat{\mathbf{x}}) \right\|^2 + w_{FB} \left\| \mathbf{y}_{FB}(\mathbf{x}(\mathbf{z}) \circ \widehat{\mathbf{x}}) \right\|^2 \right), \tag{13}$$

where the symbol '$\circ$' denotes the composition operation. For the scalars α and β, it corresponds to the addition, whereas for the homography that operation is the matrix multiplication. Furthermore, to take into account the different number of observations for IB and FB methods, we include normalization factors and define the unified output vector as

$$\mathbf{y}_{UN} = \begin{bmatrix} \sqrt{\frac{w_{IB}}{m}} \mathbf{y}_{IB} & \sqrt{\frac{w_{FB}}{2n}} \mathbf{y}_{FB} \end{bmatrix}. \tag{14}$$

Hence, a more concise unified formulation is achieved:

$$\min_{\mathbf{z}=\{\mathbf{v},\alpha,\beta\}} \frac{1}{2} \left\| \mathbf{y}_{UN}(\mathbf{x}(\mathbf{z}) \circ \widehat{\mathbf{x}}) \right\|^2, \tag{15}$$

which can be efficiently solved using [17].

C. Weight Choices

The weights w_{IB} and w_{FB} should be carefully selected to ensure the best convergence properties for the algorithm. The following constraints apply to the weights:

$$w_{IB} + w_{FB} = 1, \tag{16}$$

$$w_{FB}, w_{IB} > 0. \tag{17}$$

The idea behind the proposed method for determining the weights is to let the feature-based error be more influential to the optimization when the current solution is far from the true one. As the FB error decreases, then the intensity-based component becomes increasingly more important. This is consistent with the idea that the FB method is better suited to handle large displacements, whereas IB methods have higher accuracy, but only work when the initial guess is sufficiently close to the true solution.

The main measurement used for calculating the weights is the feature-based error associated with the current estimated homography $\widehat{\mathbf{H}}$. It is calculated using the following root mean squared error (RMSD):

$$RMSD(\mathbf{y}_{FB}) = \sqrt{\frac{\sum_{j=1}^{n} \left\| \mathbf{w}(\widehat{\mathbf{H}}, \mathbf{q}_j^*) - \mathbf{q}_j \right\|_2^2}{n}} = d_{FB}. \tag{18}$$

The proposed weights are then defined from

$$w_{FB} = 1 - \exp(-d_{FB}) \tag{19}$$

and (16). This function allows for a continuous transition where the feature-based weight decreases as its error gets lower, and the intensity-based component becomes increasingly more important in the optimization.

D. Local versus Global Search

The processing times may be drastically increased if the feature detection and matching algorithms are allowed to process the entire current image. The proposed method processes only a small region in the current image to obtain good matches whenever possible.

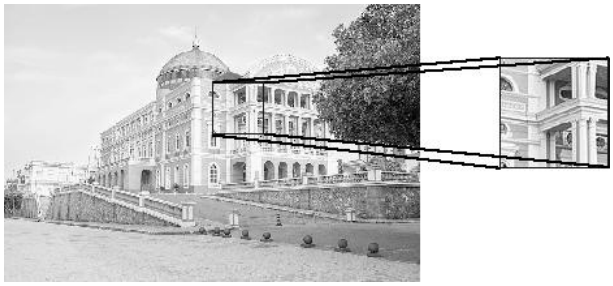
Firstly, a current template is generated by warping the current image with the initial approximation $\widehat{\mathbf{H}}$. Then, this current template is assigned a score by comparing it with the reference template using the Zero-mean Normalized Cross-Correlated. If this score is higher than a predefined threshold, then the feature detection algorithm searches only within this current template. Otherwise, the current template and $\widehat{\mathbf{H}}$ are both discarded. In this case, the detection algorithm searches the entire current image for features. The first scenario is referred to as a "local" search, whereas the second one as a "global" search. When the global search is used, it is necessary to recalculate an initial approximation $\widehat{\mathbf{H}}$. This is done by calculating the homography solely from the features matches between the current image and the reference template.

IV. Experimental Results

A. Validation Setup

The same testing procedure used in [28] is implemented to validate the algorithm. Firstly, a reference image of size 800×533 pixels is chosen, and a region of size 100×100 pixels is selected as the reference template. The coordinates

of each corner are independently perturbed in the $\overrightarrow{u}$ and $\overrightarrow{v}$ directions with a zero mean Gaussian noise and standard deviation of σ pixels (see Figure 1). The relation between the original corner points and the perturbed ones defines a test homography. The reference image is then transformed by this test homography. The algorithm receives the reference template and the transformed image with the identity element as the initial guess for the photogeometric transformation. From this input, the algorithm produces an estimated homography. In turn, this homography is used to transform each reference corner point. If the average residual error between the actual perturbed corner points and the estimated perturbed ones is less than 1 pixel, the result is declared to have converged. 1,000 test cases are randomly generated for each value of the perturbation $\sigma \in [0, 20]$ and used as input for each evaluated algorithm. In all tests, 3 levels of a multiresolution pyramid are used. In each level, a maximum of 3 iterations of the algorithm are allowed to execute.



"Teatro Amazonas Atualmente 01" by Karine Hermes | Modified

Figure 1. Validation setup. (Top) Reference image and selected reference template, resp. (Bottom) Examples of transformation with perturbations $\sigma = 5$ and $\sigma = 10$, resp.

This setup is used to compare different algorithms. Three criteria are analyzed: Convergence domain, convergence rate and timing analysis. The methods differ on whether they use only the IB or the FB component (SURF is here applied for feature detection and description) in the cost function, or both for the Unified case. Another difference is the use of a ZNCC predictor to improve the initialization in some methods. Finally, some algorithms do not consider the photometric part of the transformation space. These algorithms along with their characteristics are summarized in Table I.

TABLE I. HOMOGRAPHY ESTIMATION ALGORITHMS USED FOR COMPARISONS.

Method	IB	FB	Predictor	Photometric
ESM	✓	✗	✗	✗
IBG	✓	✗	✗	✓
IBG_P	✓	✗	✓	✓
FB_ESM	✗	✓	✗	✗
UNIF	✓	✓	✗	✓
UNIF_P	✓	✓	✓	✓

B. Convergence Domain

Figure 2 shows that the proposed Unified algorithms have a larger convergence domain than all pure FB or IB versions. It also shows that the use of the ZNCC predictor in the unified version does not affect its frequence of convergence, as well as that the IBG (i.e., IB with robustness to Global illumination changes) and ESM algorithms have a very similar performance. The latter is expected because there are no lighting changes in this validation setup.

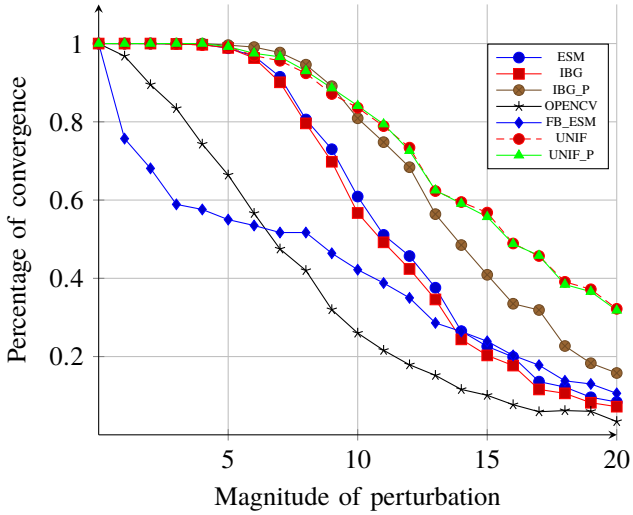


Figure 2. Percentage of convergence versus magnitude of perturbation for different homography estimation algorithms.

Another interesting observation is that the results of the algorithms in the FB class (FB_ESM and the algorithm available in OpenCV) were significantly worse than the ones in the IB class, although it was expected that they would have a higher convergence domain. This suggests that there is still room for improving the FB components of the estimation, which would in turn lead to a further improvement in the unified method as well.

C. Convergence Rate

Figure 3 compares the convergence rate of the homography estimation algorithms under a perturbation of magnitude $\sigma = 10$. This rate is displayed as the progression of the root mean squared (RMS) error between the coordinates of the 4 corners of the reference template and the estimated transformation of the current template. Out of the 1,000 test cases, only those where the estimation converged are considered here. Note that the results from the OpenCV algorithm is omitted because it was used as a black-box, and therefore the sequence of homographies at each iteration cannot be accessed. The x-axis of Figure 3 contains each important step in the optimization. The first step, which is labeled "predictor", is the result of the ZNCC prediction step. The second step, which is labeled "global", is the step where the algorithm decides to search for features in the entire current image, as described in Section III-D. Of course, these two steps are not performed by every algorithm. Afterwards, steps from the iterative optimization method follow. They are separated by pyramids level, such that the notation "X-Y" represents pyramid level X at iteration Y.

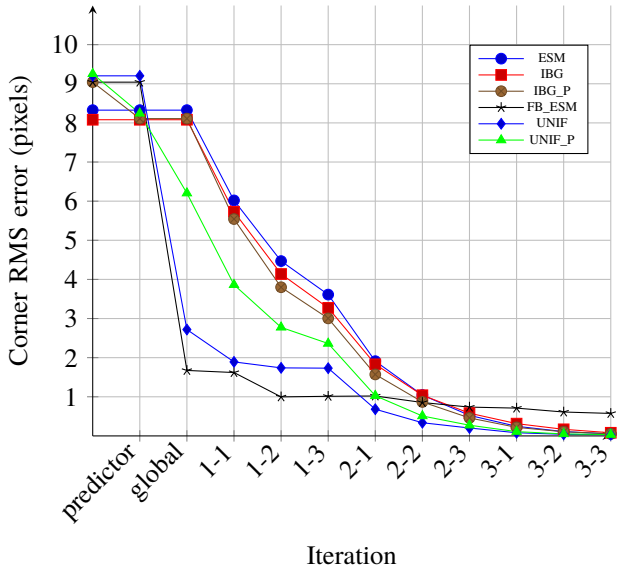


Figure 3. Pixel RMS error after each optimization iteration for different homography estimation algorithms under perturbation $\sigma = 10$.

Figure 3 allows for several observations. Firstly, the FB_ESM performance is very dependent on the "global" step. After this step, it is the algorithm with the best RMS value. However, it is not capable to improve this value too much in the subsequent optimization steps. When the other algorithms reach the third level of the pyramid, they all outperform its RMS. The behaviour of ESM, IBG and IBG_P is very similar as they share the same framework. A small difference between them is that IBG_P is able to converge even for cases with a slightly higher initial RMS error, due to the prediction step. After that step, however, all these three algorithms perform quite similarly.

Finally, let us note that the Unified algorithms have a behaviour that combines the FB and IB methods, as desired. The UNIF_P uses both the "predictor" and "global" steps. Interestingly, the global search is less applied in that version than the UNIF one because of the prediction step. This explains its smaller initial reduction in RMS value. On the other hand, less usage of the global step leads to a improvement in the processing times, as shown in the next section. After these steps, both the Unified algorithms behave similarly to IB ones, with the advantage of having a better initialization procedure.

D. Timing analysis

Figure 4 shows how the average time needed to run the estimation algorithms varies depending on the magnitude of perturbation. This time is measured in a Intel i7-6700HQ processor, and is averaged over the subset of the 1,000 cases only when the estimation has converged. The most noticeable aspect of this graph is that pure IB algorithms have nearly constant time, regardless of the perturbation level. In constrast, the algorithms that have a feature-based component need more time to process images with higher perturbation levels. This phenomenon can be explained by considering the effect of the global versus local feature search. As the perturbation level increases, the number of occasions where the algorithm applies the global search also increases. This step, however, is very computationally expensive. The UNIF_P manages to have a lower processing time because the prediction step increases the probability that the local search is used. Therefore, the UNIF_P can be seen as a compromise between having the advantage of being capable of performing global search, without taking a big penalty in the processing times.

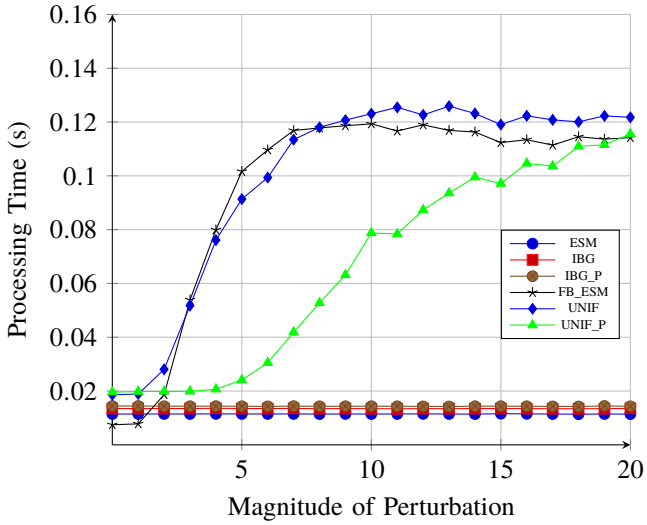


Figure 4. Processing times for different perturbation levels.

However, these results also show that more research is needed to develop a method that is able to reliably perform in real-time settings for large perturbations. The IB methods are already capable of that when they converge, requiring less than 0.02s/image. The FB and Unified methods may need up to 0.12s, which may be unacceptable for some applications.

E. Use Case: Visual Tracking

The proposed algorithm is publicly available for research purposes as a C++ library and as a ROS package [29], along with its technical report [30]. This section shows its application to homography-based visual tracking. Results are available at [31]. The prediction step is applied, as recommended for real-time tracking applications. Figure 5 shows some excerpts of this tracking experiment. An interesting result is that the proposed unified visual tracker can recover from full occlusions. Even after completely removing the tracked region from the current image, the tracker can recover given its feature-based ability to perform the "global" search. Additionally, it can be seen that the algorithm is robust to large global illumination changes, and that in some cases it can recover from complete failure even under severe lighting variations.

V. Conclusions

This paper proposes a first step towards a truly unified optimal approach to homography estimation. The results show that improved convergence properties are indeed obtained when combining both classes of feature- and intensity-based methods into a single optimization procedure. This can help vision-based applications to handle faster robot motions. Future work will focus on reducing the processing time of the unified algorithm, specially when very large interframe displacements lead to a global search for features.

 ISBN: 978-1-61208-787-0

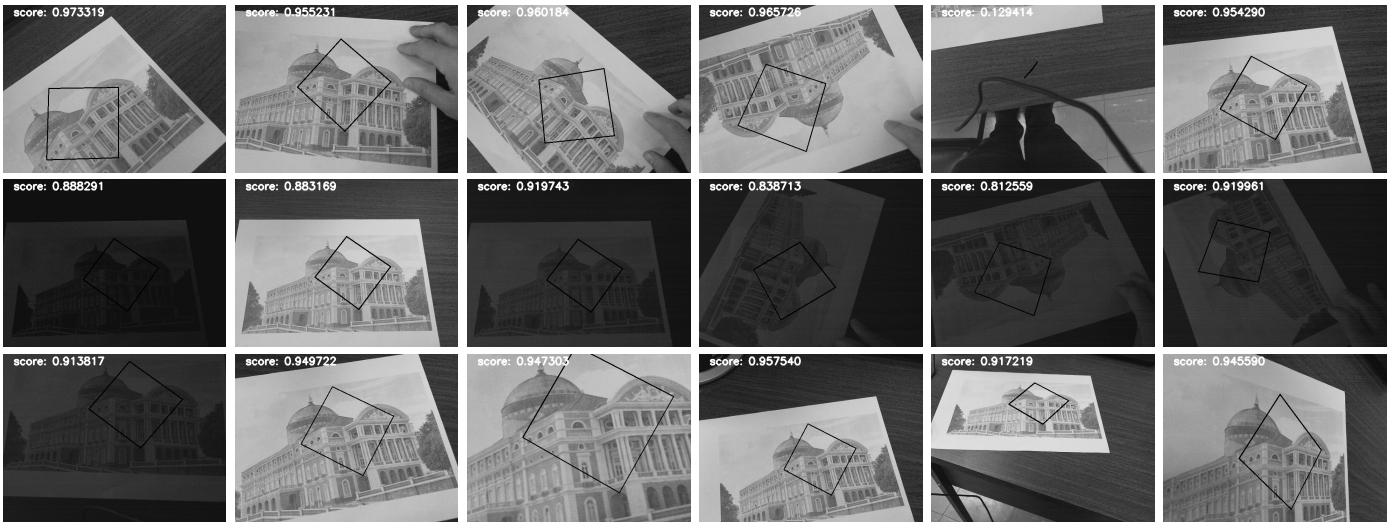


Figure 5. Excerpts of homography-based visual tracking (left-to-right then top-to-bottom) using the proposed unified approach.

Acknowledgment

This work was supported in part by the CAPES under Grant 88887.136349/2017-00, in part by the FAPESP under Grant 2017/22603-0, and in part by the InSAC project (CNPq under Grant 465755/2014-3, FAPESP under Grant 2014/50851-0).

References

[1] O. Faugeras, Q.-T. Luong, and T. Papadopoulo, The geometry of multiple images. The MIT Press, 2001.

[2] S. Benhimane and E. Malis, "Homography-based 2D visual tracking and servoing," The International Journal of Robotics Research, vol. 26, no. 7, 2007, pp. 661–676.

[3] B. Neuberger, G. Silveira, M. Postolov, and M. Vincze, "Object grasping in non-metric space using decoupled direct visual servoing," in Proc. Austrian Robotics Workshop & OAGM Workshop, 2019, pp. 99–104.

[4] L. Brown, "A survey of image registration techniques," ACM computing surveys, vol. 24, no. 4, 1992, pp. 325–376.

[5] S. Benhimane and E. Malis, "Real-time image-based tracking of planes using efficient second-order minimization," in Proc. IEEE/RSJ IROS, 2004, pp. 943–948.

[6] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: An open-source robot operating system," in Proc. ICRA workshop on open source software, 2009.

[7] R. Szeliski, "Image alignment and stitching: A tutorial," Foundations and Trends in Computer Graphics and Vision, vol. 2, no. 1, 2007.

[8] M. Irani and P. Anandan, "All about direct methods," in Proc. Workshop on Vision Algorithms: Theory and practice, 1999.

[9] G. Silveira, "Contributions to direct methods of estimation and control from visual data," Ph.D. dissertation, Ecole des Mines de Paris, 2008.

[10] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," 1981.

[11] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani, "Hierarchical model-based motion estimation," in Proc. ECCV, 1992, pp. 237–252.

[12] J. Zhang and S. Singh, "Visual-lidar odometry and mapping: Low-drift, robust, and fast," in Proc. IEEE ICRA, 2015, pp. 2174–2181.

[13] G. Silveira, E. Malis, and P. Rives, "An efficient direct approach to visual SLAM," IEEE transactions on robotics, vol. 24, no. 5, 2008, pp. 969–979.

[14] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: a versatile and accurate monocular SLAM system," IEEE Transactions on Robotics, vol. 31, no. 5, 2015, pp. 1147–1163.

[15] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Deep Image Homography Estimation," 2016.

[16] G. Silveira, "Photogeometric direct visual tracking for central omnidirectional cameras," Journal of Mathematical Imaging and Vision, vol. 48, no. 1, 2014, pp. 72–82.

[17] G. Silveira and E. Malis, "Unified direct visual tracking of rigid and deformable surfaces under generic illumination changes in grayscale and color images," International Journal of Computer Vision, vol. 89, 2010, pp. 84–105.

[18] G. D. Evangelidis and E. Z. Psarakis, "Parametric image alignment using enhanced correlation coefficient maximization," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 10, 2008, pp. 1858–1865.

[19] L. M. Fonseca and B. Manjunath, "Registration techniques for multisensor remotely sensed imagery," 1996.

[20] P. Viola and W. M. Wells III, "Alignment by maximization of mutual information," International journal of computer vision, vol. 24, no. 2, 1997, pp. 137–154.

[21] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," International journal of computer vision, vol. 56, 2004.

[22] B. Zitova and J. Flusser, "Image registration methods: a survey," Image and Vision Computing, vol. 21, 2003, pp. 977–1000.

[23] A. K. Singh, "Modular tracking framework: A unified approach to registration based tracking," Master's thesis, University of Alberta, 2017.

[24] Y. Jianchao, "Image registration based on both feature and intensity matching," in Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001, pp. 1693–1696.

[25] A. Ladikos, S. Benhimane, and N. Navab, "A real-time tracking system combining template-based and feature-based approaches," in Proc. VISAPP, 2007, pp. 325–332.

[26] P. F. Georgel, S. Benhimane, and N. Navab, "A unified approach combining photometric and geometric information for pose estimation," in Proc. BMVC, 2008.

[27] E. Malis and M. Vargas, "Deeper understanding of the homography decomposition for vision-based control," INRIA, Tech. Rep., 2007.

[28] S. Baker and I. Matthews, "Equivalence and efficiency of image alignment algorithms," in Proc. IEEE CVPR, 2001.

[29] L. Nogueira and G. Silveira, "GitHub - visiotec/vtec_ros: ROS packages from the VisioTec group," 2020, URL: https://github.com/visiotec/vtec_ros [Accessed 22 August 2020].

[30] L. Nogueira, E. de Paiva, and G. Silveira, "VTEC robust intensity-based homography optimization software," no. CTI-VTEC-TR-01-19, Brazil, 2019.

[31] L. Nogueira, "Unified Intensity- And Feature-Based Homography Estimation Applied To Visual Tracking," 2020, URL: https://www.youtube.com/watch?v=oArw449qp1E [Accessed 22 August 2020].

 ISBN: 978-1-61208-787-0