Is Multi-Model Feature Matching Better for Endoscopic Motion Estimation?

Xiang Xiang, Daniel Mirota, Austin Reiter, Gregory D. Hager

Dept. of Computer Science, Johns Hopkins University, Baltimore, MD 21218, USA
{xxiang,dan,areiter,hager}@cs.jhu.edu

Abstract. Camera motion estimation is a standard yet critical step to endoscopic visualization. It is affected by the variation of locations and correspondences of features detected in 2D images. Feature detectors and descriptors vary, though one of the most widely used remains SIFT. Practitioners usually also adopt its feature matching strategy, which defines inliers as the feature pairs subjecting to a global affine transformation. However, surfaces are non-planar in endoscopic videos. We are curious if it is more suitable to cluster features into multiple groups. We can still enforce the same transformation as in SIFT within each group. Such a multi-model idea has been recently examined in the Multi-Affine work, which outperforms Lowe's SIFT in terms of re-projection error on minimally invasive endoscopic images with manually labelled ground-truth matches of SIFT features. Since their difference lies in matching, the accuracy gain of estimated motion is attributed to the holistic Multi-Affine feature matching algorithm. But, more concretely, the matching criterion and point searching can be the same as those built in SIFT. We argue that the real variation is only the motion model verification. We either enforce a single global motion model or employ a group of multiple local ones. In this paper, we investigate how sensitive the estimated motion is affected by the number of motion models assumed in feature matching. While the sensitivity can be analytically evaluated, we present an empirical analysis in a leaving-one-out cross validation setting without requiring labels of ground-truth matches. Then, the sensitivity is characterized by the variance of a sequence of motion estimates. We present a series of quantitative comparison such as accuracy and variance between Multi-Affine motion models and a global affine model used in SIFT.

1 Introduction

It is estimated that there are more than 200,000 functional endoscopic sinus surgeries (FESS) procedures performed annually in US. As the name implies, all of these procedures performed via endoscopic visualization, and a large fraction employ surgical navigation systems to visualize critical structures that must not be disturbed during the surgery. Although navigation is widely employed for FESS, its capabilities are far from optimal. In particular, the sinuses contain structures that are smaller than a millimeter in size, and yet delineate critical anatomy such as the optic nerve or the carotid artery. However, the accuracy of navigation is 2 mm under near ideal conditions. As a result, navigation can provide a qualitative sense of location, but final confirmation of anatomic structures ultimately relies on the surgeon's ability to interpret and relate the CT

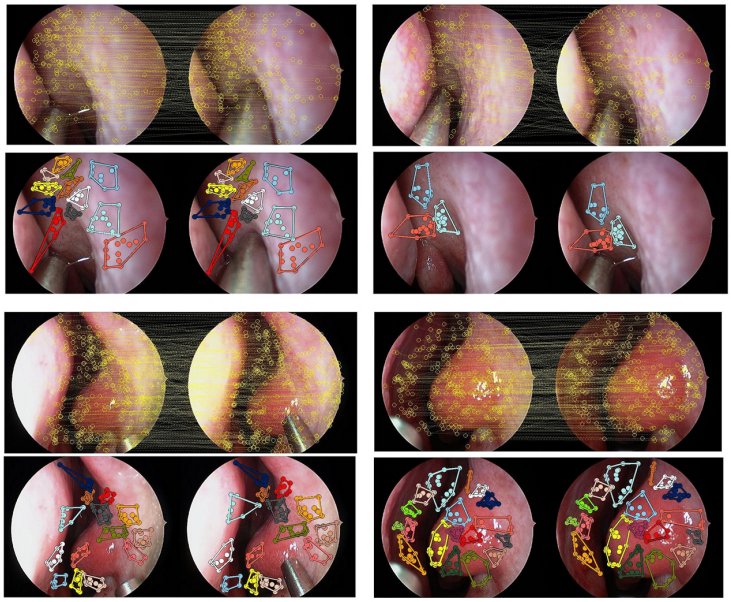

Fig. 1. Examples of SIFT's global-affine vs. HMA's multi-affine model. In each pair, the top row shows SIFT's result, in which line crossings imply mismatches. The bottom row shows HMA's result, in which different components are displayed in different color.

image to the endoscopic view. This process, which is further complicated when the anatomy is distorted or otherwise altered by surgery, requires time, skill and experience and can lead to errors in judgement that adversely affect outcome.

The significance of endoscopic visualization [1, 2] is inducing a paradigm shift in surgical navigation by using endoscope to improve anatomy registration. It provides an inexpensive, non-invasive, radiation-free method to enhance registration accuracy at any point of the procedure. In a big picture, the pipeline consists of feature processing, motion estimation, tracking and 3-D reconstruction [2]. In this paper, we focus on the **motion estimation** which interacts with **feature matching**. While the global camera motion is estimated from matched features, we first need a preliminary motion model to verify the feature matches. Subsequently, [3] shows the limitations of image-based tracking alone can be overcome by employing an Electro-Magnetic (EM) tracker, which provides a rough location. EM tracking can correct drifting, while frame-by-frame tracking-by-matching gives a refined location. Lastly, reconstruction can follow a point cloud generation by either simple triangulation [2, 4] or bundle adjustment [5, 6], or surface rendering methods using shading [1] and specularities [7].

In such a pipeline of 3-D visualization, the camera motion estimation is critical to the final accuracy. It is standard to estimate the global motion once feature matches are available. Eight-point algorithm [4] or the relaxed five-point algorithm [8] give quite similar estimates. [9] can even handle the wide-baseline problem. Then, the variation comes from previous steps such as feature detec-

tion, description and matching. The Scale Invariant Feature Transform (SIFT) [10] is invariant to image scaling and rotation, and partially invariant to changes in illumination and 3D camera viewpoint. We would like to fix the detector and descriptor to be SIFT and then concentrate on examining **how sensitive the estimated motion is affected by the feature matching**.

In this paper, we compare the matching algorithm built in the original SIFT [10] with the state-of-the-art Multi-Affine matching [11–13] (typically the Hierachical Multi-Affine (HMA) [12]). In detail, feature matching consists of deciding the matching criterion, searching a similar feature and verifying if the matches agree with the motion model [14]. Firstly, SIFT's matching strategy is thresholding the ratio of nearest/2nd-nearest Euclidean distance in the feature space. HMA follows that as well. Secondly, the variation of searching algorithms highlighted in [12] is about efficiency. Thirdly, matching is normally expected to be robust to outliers and deformation. Therefore, a model verification step aims to check the agreement between each feature and the motion model. However, HMA and the SIFT differ in the number of motion models. It is entirely possible to replace the affine model built in SIFT with any linear or non-linear model. However, we would rather fix it to be affine and then can investigate **if multi-model is really superior over single-model for feature matching**.

As a result, this paper attempts to illustrate **how sensitive the estimated motion is affected by the number of motion models used in feature matching**. Surely we can approximately analyze the sensitivity of estimated motion $[\mathbf{R}, \mathbf{t}]$ by applying a direct differentiation of the epipolar constraint [15] and linearzing the motion displacement $[\Delta\mathbf{R}, \Delta\mathbf{t}]$. We can explicitly write down the correlation between the covariance of $[\mathbf{R}, \mathbf{t}]$ and the variance of the matched feature pairs [16]. However, it is still significant to verify such an approximation with the empirical sensitivity by estimating the motion a number of times. In this way, we can also compute the accuracy of feature matching in terms of inlier ratio and the accuracy of estimated motion in terms of re-projection error.

2 Empirical Sensitivity of Motion Estimation

In this section, we quantitatively analyze motion estimation's sensitivity, which is captured in the variance of the estimated rotation [6]. The basic idea is to generate a sequence of estimate by Leaving-One-Out Cross Validation (LOOCV). In each trial, we use one feature pair for querying and the remaining for estimating the motion. In this way, we can validate how the holistic motion estimation pipeline generalizes such as robustness without requiring many batches of data. The less sensitively an model behaves on real data, the more robust it is.

For feature matching, the input are features and a motion model, while the output are feature pairs and motion model parameters. For motion estimation, the input are feature pairs and the output are motion model parameters. Note that the two motion models can be different. The former is a preliminary model assumed to characterize a reasonable transformation between two images, which are not arbitrary two images though. They characterize adjacent views which approximately capture the same scene from a monocular camera. Thus, the

latter model subjects to a more strict geometric constraint called the epipolar constraint [4]. It is normally affine as well, with a rotation and a translation. And its estimation method is quite standard such as the five-point algorithm [8] we use. The only variation is the design of the preliminary motion model. SIFT uses a global affine model while HMA uses a local affine model for each component, which is obtained through a hierarchical K-means clustering and expected to represent a plane [13]. The way to estimate the model parameters is once again rather standard. In both cases, the model parameters are computed by solving a linear system from feature pairs for the model parameters. Then, they are refined by verifying the agreement bewteen the raw feature pairs and the parameterized model based on some robust fitting methods, such as the Hough transform that SIFT uses and the RANdom SAmple Consensus (RANSAC) that we use. We use RANSAC for both HMA and SIFT. Its basic idea is to iteratively use a minimal number of pairs needed to re-estimate the model [4]. **Inliers** are defined as the pairs subjecting to the refined motion model. While inliers are detected separately for each group in HMA, we will compare its holistic inlier ratio with SIFT. Also note that one variation in our sensitivity analysis is the representation of a rotation. Instead of quaternions [17], another way is to use Euler angles [18]. When rotation angles are small, both ways can be an alternative to the rotation matrix [19]. Finally, the exact algorithm is elaborated in Algorithm 1.

Algorithm 1. Sensitivity analysis of motion estimation by LOOCV.
for $k = 1...FrmNum$
  Form a candidate pool: detect SIFT keypoints and compute SIFT features.
  Match features: fit feature pairs to a single-affine **or** multi-affine model.
  Generate an inlier set: perform RANSAC on the matched keypoint pairs.
  Rectify images considering radial distortion.
  Convert image coordinates to World's coordinates using intrinsic parameters.
  if $MatchedInlierNum > 4$
    for $trial = 1...MatchedInlierNum$
      Leave the $trial$-th keypoint pair $(\mathbf{p}_{left}^{query}, \mathbf{p}_{right}^{query})$ out as a query.
      Estimate the essential matrix $\mathbf{E}$ using the remaining pairs.
      Factorize $\mathbf{E}$ into a rotation matrix $\mathbf{R}$ and a translation vector $\mathbf{t}$.
      Convert $\mathbf{R}$ to a quaternion **or** yaw, pitch and roll angles rx, ry, rz.
      Compute square of re-projection error for the held-out query point:
        $sqErr = \|(\mathbf{R} * \mathbf{p}_{left}^{query} + \mathbf{t}) - \mathbf{p}_{right}^{query}\|_2^2$
    end for
    Compute the mean and standard deviation of a sequence of $sqErr$.
    Compose a $\mathbf{R}_{mean}$ from $mean(quaternion)$ **or** $mean(rx), mean(ry), mean(rz)$.
    for $trial = 1...MatchedInlierNum$
      $\mathbf{R}_{mean} * \mathbf{R}^{-1}$ is approximately a skew-symmetric matrix $skew(\alpha, \beta, \gamma)$
    end for
    Compute the standard deviations of rotation angles α, β, γ, respectively.
  end if
end for

Now, some terminologies appeared are informally explained in the following. **Affine motion model** restricts the motion of each point to depend linearly on its location (*ie*, a linear model) plus a constant offset (*ie*, a translation) [4]. **Motion model verification** uses the motion model to validate the consistency of feature pairs (left point, right point). Conversely, feature matches are used for **solving the model parameters**. We can use all the parameters to form a column vector $\mathbf{x}$, Correspondingly, we encode locations of the left and right points in a matrix $\mathbf{A}$ and a column vector $\mathbf{b}$. Then, solving for the model parameters $\mathbf{x}$ becomes fitting the point pairs $\mathbf{A}$ and $\mathbf{b}$ to the parameterized model $\mathbf{Ax} = \mathbf{b}$. The solution is given by minimizing a certain error metric such as the sum of square residues: $\mathbf{x} = \arg\min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_2^2$. Since an affine model can be characterized using 6 parameters, the linear system is over-determined as long as there are over 6 point pairs. Although the exact solution may not exist, it is unique if existing. Moreover, a closed-form approximate solution can always be given by Least Squares: $\mathbf{x} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$. Even if we use more complicated models with many parameters, the linear system can be under-determined in the case with a few point pairs. $\mathbf{x}$ can be seen as a dimension-augmented representation of $\mathbf{b}$. If we insist in applying Least Squares, augmented dimension will be hallucinated. But the data in $\mathbf{A}$ are redundant, so we can seek a **sparse** usage of $\mathbf{A}$ by adding a constraint $\|\mathbf{x}\|_1 \leq T$, which is relaxed from $\|\mathbf{x}\|_0 \leq S$. **Coordinate transformation** from image coordinates' to World's coordinates follows the standard geometric model of image formation [4] as shown below. In the *r.h.s*, the first matrix encodes the scaling information and the second

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_x & 0 & 0 \\ 0 & f_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ I \end{bmatrix}$$

encodes the focus. The product of these two matrices is termed the intrinsic parameter matrix. Here the projection matrix is not included since the original coordinates are uncalibrated 2D image coordinates, instead of World's 3D coordinates. However, only linear distortion is considered in this equation. The **radial distortion** has initially first compensated by image rectification [4]. **Essential matrix E** is a 3×3 matrix encoded in the epipolar constraint and be decomposed to the relative pose/motion $[\mathbf{R}, \mathbf{t}]$ based on the SVD of $\mathbf{E}$ [4]. **Quarternion of a rotation**. Unit quarternion [17] is a four-element vector. A rotation matrix can be represented by a quarternion as:

$$\boldsymbol{R} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 + q_2^2 - q_1^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 + q_3^2 - q_1^2 - q_2^2 \end{bmatrix}$$

Euler angles of a rotation [18] are computed according to Euler's rotation theorem [20], which implies that the composition $\Delta\mathbf{R}$ of two rotations $\mathbf{R}_{mean}$ and $\mathbf{R}^{-1}$ is also a rotation. From [20], suppose we specify an axis of rotation by a unit vector $[x, y, z]$ and we have an infinitely small rotation of angle $\Delta\theta$ about

the vector. Expanding the rotation matrix as an infinite addition, and taking the first-order Taylor series expansion, the rotation matrix $\Delta\mathbf{R}$ is represented as

$$\Delta R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & z & -y \\ -z & 0 & x \\ y & -x & 0 \end{bmatrix} \Delta\theta = \mathbf{I} + \mathbf{A}\,\Delta\theta.$$

Note that $\Delta\mathbf{R}$ is a skew symmetric matrix, where the element x, y, z denotes the Euler angle in X, Y, Z axis, respectively. In Algorithm 1, x, y, z correspond to α, β, γ. Namely, α, β, γ are the rotation angle in X, Y, Z axis, respectively.
Re-projeection error is a geometric error, which is the distance between a projected point and a measured one [6]. We project the held-out query keypoint using the estimated $[\mathbf{R}, \mathbf{t}]$ and compute its Euclidean distance to its pair.

3 Experiments

In this section, we present the experimental results of the feature matches' accuracy, the estimated motion's accuracy, and the estimated motion's sensitivity with respect to the number of motion model used in feature matching.

3.1 External Libraries

Camera calibration is performed by using Caltech calibration toolkit [21].
SIFT features are extracted using VLfeat library [22].
HMA matching strategy is performed using HMA toolbox [23].
Camera motion recovery from $\mathbf{E}$ is via Structure and Motion toolkit [24].

3.2 Patient Data

Fig. 2 shows the data collection system we developed to simultaneously capture both the endoscopic video and EM tracking data, though we do not examine EM data in this paper. The video that we have collected lasts for hours at 30 FPS. As summarized in Fig. 2, we down-sample the sequence to be at 1 FPS and select 64 continuous frames, among which 46 frames are with over 4 SIFT keypoints detected. The baseline between lens in two monocular adjacent views

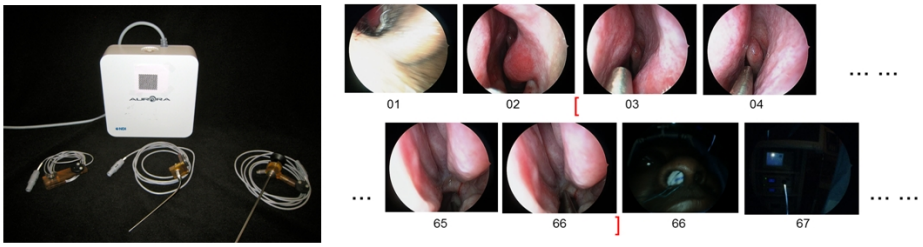

Fig. 2. The left figure shows the endoscopic sensor and data collection devices. The top box is the processor from NDI. The bottom left is a high-precision optically tracked endoscope, The bottom middle and right form a EM tracked scope for use in airway data collection. The right figure shows an endoscopic video sample of a patient's sinus.

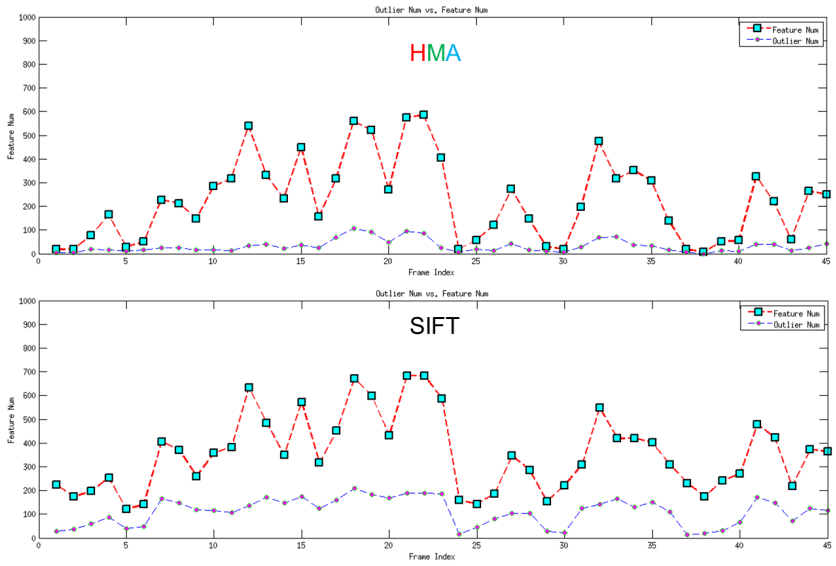

Fig. 3. RANSAC detected **outlier** number vs. Total matches number. HMA generates much fewer (< 100) outliers than SIFT does (> 100). Better seen on computer.

is relatively small. Interested readers may refer to [2] for how sensitive is the estimated motion affected by the baseline. Endoscopic images normally contain scaling and rotation, changes in illumination and 3D camera viewpoint, and low-textured non-planar surfaces. Usually a few feature keypoints are detected. Then, the hope lies in a large inlier ratio of matched keypoint pairs. In a certain frames such as frame 03 and 04, the partial occlusion is mainly from the tools and specularities are mostly a result of air bubbles forming on local wet surfaces.

3.3 Accuracy of Feature Matching

Between HMA and SIFT, Fig. 1 presents a qualitative comparison of matched features. SIFT presents a number of line crossing which imply mismatches. For instance, a SIFT keypoint in the left region can be matched to another in the right region, which is unlikely to be given by HMA. This is verified in Fig. 3 that presents a quantitative comparison of the detected outlier number given by RANSAC verification vs. the total matched feature number given by the tentative matching. We can see that although HMA and SIFT generate a similar number of matched features, HMA induces a higher inlier ratio than SIFT does.

3.4 Accuracy of Estimated Motion

Now, we project the held-out query pair using the estimated $[\mathbf{R}, \mathbf{t}]$. For each pair of adjacent frames, the input are locations of keypoints in the frame τ and those in the frame $\tau + 1$, together with the essential matrix $\mathbf{E}$. The Mean Square

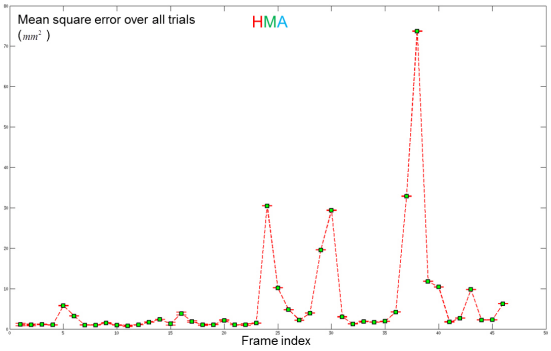

Fig. 4. Re-projection error of the held-out query keypoint with HMA matching.

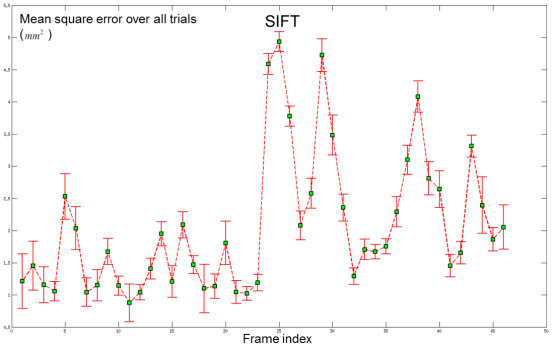

Fig. 5. Re-projection error of the held-out query keypoint with SIFT matching.

Error (MSE) for HMA and SIFT are presented in Fig. 4 and Fig. 5, respectively. In both cases, the majority frames' MSE are within 5 $pixel^2$, which verifies both HMA and SIFT matches are generally accurate. The next question is how sensitive is the estimated motion affected by the feature matching.

3.5 Sensitivity of Estimated Motion

Now, we will examine the estimated rotation $\mathbf{R}$. The variances of the rotation angle α, β, γ characterize the sensitivity of the rotation. Please review Algorithm 1 for computing the rotation angles. Fig. 6 displays the respective standard deviation together with the number of features for both HMA and SIFT. **Over time the standard deviation curves of rotation angles follow similar trends, which are generally in the opposite direction of the feature number curve.** Namely, when there are more feature matches, the rotation angles are less variant. Notably, the difference of feature number is important only when the features are relatively a few, which is more or less the case in endoscopic images. Moreover, the number of well-matched features rely on the matching algorithm, Thus, for low-textured scenes such as sinuses with textureless surfaces and specularities, we can somewhat conclude that the estimated motion is sensitive to the number of motion models employed in feature matching.

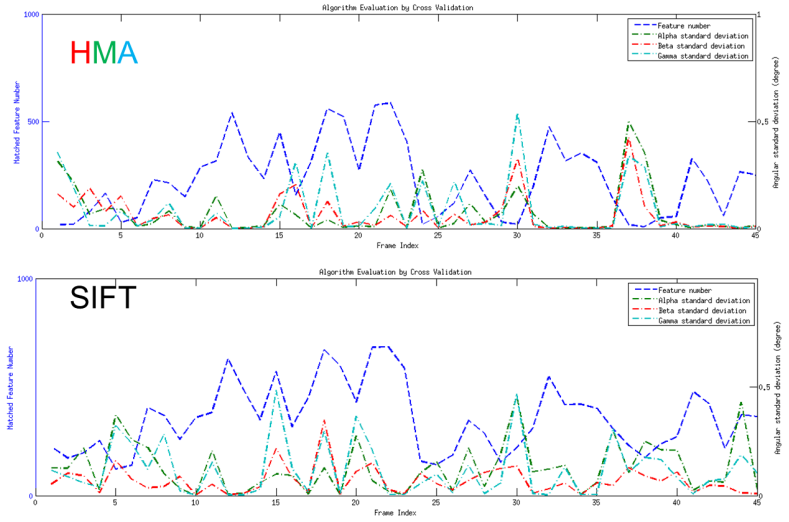

Fig. 6. The standard deviation of α, β, γ vs. feature number. The left Y-axis denotes the matched feature number, which is displayed in blue. The right Y-axis denotes the angular standard deviation. Better to be seen on computer.

4 Discussion

In this paper, we are interested in three questions. Firstly, does multi-model induce more accurate feature matches than single-model? Secondly, is the camera motion estimated from feature matches given by multi-model more accurate than those given by single-model? Thirdly, how sensitive is the estimated motion affected by the motion model number in feature matching? We empirically investigate those questions by conducting cross validation on endoscopic videos of sinuses. We find that although multi-model (HMA) and a single-model (the original SIFT) generate a similar number of matched features, HMA induces a higher inlier ratio than SIFT does. Besides, both HMA and SIFT matches are generally accurate, for the majority frames' MSE are within 5 $pixel^2$ in both cases. Moreover, when there are more feature matches, the rotation angles are less variant. It indeed verifies that the estimated motion in low-textured endoscopic scenarios is sensitive to the number of motion model used in feature matching.

ACKNOWLEDGMENTS

This work is supported by US Natl. Inst. of Health under grant R01 EB015530. The first author is grateful for the fellowship from China Scholarship Council.

References

1. Tokgozoglu, H.N., Meisner, E.M., Kazhdan, M., Hager, G.D.: Color-based hybrid reconstruction for endoscopy. In: CVPR Workshops. (2012)

2. Mirota, D., Wang, H., Taylor, R.H., Ishii, M., Gallia, G.L., Hager, G.D.: A system for video-based navigation for endoscopic endonasal skull base surgery. IEEE T-MI **31** (2012) 963–976
3. Mori, K., Deguchi, D., Akiyama, K., Kitasaka, T., Maurer, C.R., Suenaga, Y., Takabatake, H., Mori, M., Natori, H.: Hybrid bronchoscope tracking using a magnetic tracking sensor and image rgistration. In: MICCAI. (2005)
4. Ma, Y., Soatto, S., Kosecka, J., Sastry, S.: An Invitation to 3-D Vision. Springer (2004)
5. Wu, C.: VisualSFM: A Visual Structure from Motion System. `http://ccwu.me/vsfm/` (2011)
6. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Second edn. Cambridge University Press (2004)
7. Collins, T., Bartoli, A.: Towards live monocular 3d laparoscopy using shading and specularity information. In: IPCAI. (2012)
8. Nister, D.: An efficient solution to the five-point relative pose problem. IEEE Trans. Pattern Analysis and Machine Intelligence **26** (2004) 756–770
9. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide-baseline stereo from maximally stable extremal regions. In: BMVC. (2002)
10. Lowe, D.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60** (2005)
11. Puerto-Souza, G.A., Mariottini, G.L.: Adaptive multi-affine (ama) feature-matching algorithm and its application to minimally-invasive surgery images. In: MICCAI. (2012)
12. Puerto-Souza, G.A., Mariottini, G.L.: Hierarchical Multi-Affine (HMA) algorithm for fast and accurate feature matching in Minimally-Invasive surgical images. In: IEEE IROS. (2012)
13. Puerto-Souza, G.A., Mariottini, G.L.: A fast and accurate feature-matching algorithm for minimally invasive endoscopic images. IEEE T-MI (2013)
14. Szeliski, R.: Computer Vision: Algorithms and Applications. Springer (2010)
15. Abretske, D., Mirota, D., Hager, G.D., Ishii, M.: Intelligent frame selection for anatomic reconstruction from endoscopic video. In: WACV. (2009)
16. Mirota, D.: Video-Based Navigation with Application to Endoscopic Skull Base Surgery. Johns Hopkins University Computer Science Ph.D. Dissertation (2012)
17. Wikipedia: Quaternion. (`http://en.wikipedia.org/wiki/Quaternion`)
18. Wikipedia: Euler Angles. (`http://en.wikipedia.org/wiki/Euler_angles`)
19. Wikipedia: Rotation Formalisms in Three Dimensions. (`http://en.wikipedia.org/wiki/Rotation_formalisms_in_three_dimensions`)
20. Wikipedia: Euler's Rotation Theorem. (`http://en.wikipedia.org/wiki/Euler's_rotation_theorem`)
21. Caltech Vision Lab: Camera Calibration Toolbox for Matlab. `http://www.vision.caltech.edu/bouguetj/calib_doc/` (2010)
22. Vedaldi, A., Fulkerson, B.: VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/` (2008)
23. Puerto, G.A., Mariottini, G.L.: HMA Feature-Matching Toolbox. `http://ranger.uta.edu/~gianluca/feature_matching/` (2012)
24. Torr, P.: Structure and motion toolkit. `http://www.mathworks.com` (2004)