2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Robust Jacobian Estimation for Uncalibrated Visual Servoing

Azad Shademan, Amir-massoud Farahmand, and Martin Jägersand

Abstract— **This paper addresses robust estimation of the uncalibrated visual-motor Jacobian for an image-based visual servoing (IBVS) system. The proposed method does not require knowledge of model or system parameters and is robust to outliers caused by various visual tracking errors, such as occlusion or mis-tracking. Previous uncalibrated methods are not robust to outliers and assume that the visual-motor data belong to the underlying model. In unstructured environments, this assumption may not hold. Outliers to the visual-motor model may deteriorate the Jacobian, which can make the system unstable or drive the arm in the wrong direction. We propose to apply a statistically robust M-estimator to reject the outliers. We compare the quality of the robust Jacobian estimation with the least squares-based estimation. The effect of outliers on the estimation quality is studied through MATLAB simulations and eye-in-hand visual servoing experiments using a WAM arm. Experimental results show that the Jacobian estimated by robust M-estimation is robust when up to 40% of the visual-motor data are outliers.**

I. Introduction

In unstructured environments, the motion control of vision-based robotic systems must be independent from geometric structure, model, and calibration parameters. Vision-based motion control has been long studied in visual servoing, where visual information is used to control a robot to a desired configuration by minimizing an error norm associated with a task [1]. Visual tracking errors often play a key role in the failure of a visual servo, therefore, robustness to tracking errors should also be considered.

In this paper, we propose to apply a robust M-estimator to numerically estimate the visual-motor Jacobian from raw visual-motor data and statistically reject the outliers due to different visual tracking errors. Our proposed method does not require any prior knowledge on neither camera/robot calibration, nor the geometric model of features. The block diagram of our proposed system is depicted in Figure 1.

Classical approaches to visual servoing use some knowledge about the model and/or the system parameters [2]. For example, position-based visual servoing (PBVS) [3] uses a calibrated camera and known geometric model of the 3D features to reconstruct the relative pose of the camera with respect to the desired object. In this sense, PBVS is *model-based* and not suitable for unstructured settings. The classical image-based visual servoing (IBVS) [4] does not require the geometric model, but the analytic form of the image Jacobian is often used in the control loop. This image Jacobian contains the intrinsic camera parameters and often a 3D parameter expressed in the camera frame (*e.g.,* for point features, this 3D parameter is the depth). In this sense, IBVS depends on a parametric model of the image Jacobian which must be derived analytically in advance for each feature type. Therefore, we consider classical IBVS as a model-based approach [1]. An issue with the classical IBVS approach is that the 3D parameter(s) cannot be directly measured and must be estimated from images. In addition, implementation of the traditional PBVS and IBVS requires the knowledge of the extrinsic calibration of the camera with respect to the end-effector.

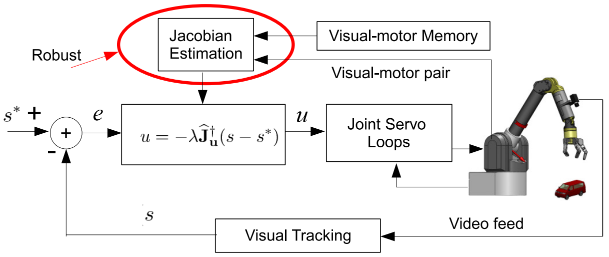

Fig. 1. System block diagram. The main contribution of this paper is the robust Jacobian estimation for the uncalibrated IBVS system (see Section III). A WAM with eye-in-hand configuration is considered.

Advanced visual servoing approaches have been proposed to address some of the above issues [6]. One of these advanced methods is called uncalibrated visual servoing, which is an IBVS approach without the need for the scene model or camera/robot calibration [7]–[9]. The uncalibrated image Jacobian relates the joint velocity directly to image-space velocities, which can be estimated from previous measurements of visual-motor data. This sets the system free from any models or parameters. In this sense, uncalibrated visual servoing in [7]–[9] is *model-free* and *nonparametric* . Such uncalibrated methods are particularly important when the analytic form of the Jacobian is not available or tedious to derive. Throughout this paper, we use the *uncalibrated Jacobian* in this nonparametric model-free sense.

Robustness in visual servoing has been studied from different perspectives: robust visual tracking, statistically robust IBVS, and control-theoretic stability analysis using adaptive robust control. Many authors discuss robust visual tracking for a visual servoing system. Kragic and Christensen [10], [11] propose robust visual tracking using a

This work is supported by NSERC, CFI, iCORE, and the Alberta Advanced Education & Technology. A. Farahmand acknowledges the support of AICML.

Authors are with the Department of Computing Science, University of Alberta, Edmonton, AB, T6G2E8, Canada. {azad,amir,jag}@cs.ualberta.ca

[1] Some authors refer to IBVS as model-free in the sense that it does not depend on the geometric model of the object (*e.g.,* [5]). This should not be confused with our usage of model-based which refers to a parametric Jacobian model.

978-1-4244-5040-4/10/$26.00 ©2010 IEEE

voting scheme and visual cue integration to achieve robustness in unstructured settings. Preisig and Kragic [12] use robust M-estimation and RANSAC for 3D tracking. Comport *et al.* [13] compare statistically robust real-time visual tracking algorithms. Tran and Marchand [14] propose a fast and efficient feature descriptor for tracking and use RANSAC to reject matching outliers between two views. Robust M-estimation has been proposed for articulated object tracking with applications to virtual reality by Comport *et al.* [15]. Another set of papers discuss statistical robustness within the image-based control law [16], [17]. More closely related to the proposed method is the work of Comport *et al.* [16], which proposes a statistically robust IBVS control law. Nonetheless, they use a parametric Jacobian, where robustness is achieved by down-weighting the rows of the parametric Jacobian associated with *corrupted* features. Thus, our proposed method differs from the work of Comport *et al.* [16] in that we directly exploit the statistics of the visual-motor data to estimate the robust Jacobian (see Figure 1).

Other approaches to robust visual servoing are either model-based [5], [18], [19], or model-free but parametric [20], [21]. The latter, study the problem from an adaptive control-theoretic point of view and estimate the linearized camera/robot calibration parameters using a depth-independent Jacobian without estimating the depth directly. Wang *et al.* [20] propose a depth-independent Jacobian to estimate the linearized camera parameters on line. They use this depth-independent Jacobian for eye-in-hand visual servoing of point and line features without knowledge of 3D coordinates. Hu *et al.* [21] propose a homography-based robust adaptive controller to control translation and orientation of an eye-in-hand system in the presence of uncertainty in the intrinsic camera parameters as well as uncertainty in the depth information. These methods do not study robustness to visual-motor outliers.

II. Background

A. Uncalibrated IBVS

The control law in uncalibrated visual servoing is defined entirely in the image space without the need to reconstruct the depth or other 3D parameters.

Formally, let $F : \mathbb{R}^N \rightarrow \mathbb{R}^M$ be the mapping from configuration $q \in \mathbb{R}^N$ of a robot with N joints, to the visual feature vector $s \in \mathbb{R}^M$ with M visual features, *i.e.,* $s = F(q)$ is the visual-motor function of the robotic hand/eye system (see Figure 1). The time derivative of the visual-motor function leads to the visual-motor Jacobian $\mathbf{J_u}$:

$$\begin{aligned} \frac{\partial s}{\partial t} &= \frac{\partial F(q)}{\partial q}\frac{\partial q}{\partial t}, &\quad (1)\\ \dot{s} &= \mathbf{J_u}(q)\dot{q}. &\quad (2) \end{aligned}$$

If an estimate $\widehat{\mathbf{J}}_\mathbf{u}(q)$ for $\mathbf{J_u}(q)$ is available, the discrete-time form of (2) becomes

$$\Delta s \simeq \widehat{\mathbf{J}}_\mathbf{u}(q)\Delta q. \quad (3)$$

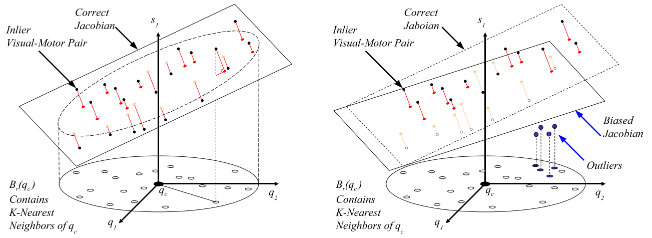

Fig. 2. (Left) Illustration of the Local Least-Squares (LLS) method. (Right) Error caused in hyperplane estimation due to outliers.

To reach a visual goal s^*, the estimated Jacobian $\widehat{\mathbf{J}}_\mathbf{u}$ is used in a control law, *e.g.,*

$$\dot{q} = -\lambda\widehat{\mathbf{J}}_\mathbf{u}^\dagger(s - s^*), \quad (4)$$

where $\widehat{\mathbf{J}}_\mathbf{u}^\dagger$ is the MoorePenrose pseudoinverse of $\widehat{\mathbf{J}}_\mathbf{u}$. In the remainder of this section, we review a background to the estimation of the uncalibrated Jacobian.

B. Visual-Motor Jacobian Estimation

A Broyden rank-one secant update has been proposed by Jägersand *et al.* [8] and Hosoda and Asada [7] to estimate the visual-motor Jacobian. For critical tasks, a good initial guess is essential. Usually, methods based on [7], [8] use a forgetting factor to lessen the weight of old data during the estimation process. However, old data are useful. Nevertheless, methods do not incorporate nearby points which have been visited in the past.

Farahmand *et al.* [9] propose local least-squares (LLS) estimation to utilize the memory of visual-motor data. They estimate the visual-motor Jacobian in simulated 3 degrees-of-freedom (DOF) eye-to-hand experiments. This method is general and estimates the Jacobian of any point in the workspace directly from raw visual-motor data in a close neighborhood of the point under consideration. We compare our method to theirs.

For a memory with P visual-motor data pairs and a new visual-motor query point $d_c = (s_c, q_c)$, the uncalibrated Jacobian estimation problem is posed as the following optimization problem [9]:

$$\widehat{\mathbf{J}}_\mathbf{u}(q)\Big|_{q=q_c} = \arg\min_{\mathbf{J_u}} \sum_{k:\, q_k \in B_r(q_c)} (\Delta s_k - \mathbf{J_u}\Delta q_k)^2, \quad (5)$$

where $B_r(q_c) = \{q_p : \|q_c - q_p\| < r\,, p = 1, \cdots, P\}$ is an open ball with radius r centered at q_c which contains joint-space neighbors of query joint q_c, $\Delta s_k = s_c - s_k$, and $\Delta q_k = q_c - q_k$. This method fits the best hyperplane to the visual-motor data around q_c. An illustration of this method is given in Figure 2 (Left), where a hyperplane is fitted to 2×1-dimensional data (2 DOF for joints and a single image feature). Outliers can cause errors in estimation. The erroneous fitted model is shown by a biased plane in Figure 2 (Right). We will present a statistically robust method to deal with outliers in Section III. The LLS method [9] is similar to the work of Lapresté *et al.* [22], where the least squares problem is solved directly for the pseudoinverse

Jacobian. However, while LLS [9] is on-line, off-line training by random perturbations around the desired pose are used in [22].

III. Robust Jacobian Estimation

The least squares methods are sensitive to outliers in the visual-motor data. Examples of outliers are occlusion of visual features being tracked, visual tracking mismatches (mistracking), or large visual tracking errors. In the presence of such visual tracking discrepancies, the corresponding visual-motor pair (s, q) are outliers and do not belong to the model being fit. Therefore, the performance of least squares-based methods deteriorates. The influence of outliers needs to be removed during the estimation. Robust regression provides a proper mathematical framework to deal with outliers. Here, we apply some widely accepted robust statistics and estimators to estimate the uncalibrated Jacobian.

A. Problem Formulation

Denote a memory of P visual-motor data pairs (s_p, q_p) by set $D = \{(s_p, q_p)\}_{p=1}^{P}$. Given a visual-motor query point $d_c = (s_c, q_c) \notin D$, consider estimating the Jacobian in (3) at d_c by minimizing the sum of a robust norm of the residuals:

$$\widehat{\mathbf{J}}_{\mathbf{u}}(q)\Big|_{q=q_c} = \arg\min_{\mathbf{J}_{\mathbf{u}}} \sum_{k:\, q_k \in B_r(q_c)} \rho(\Delta s_k - \mathbf{J}_{\mathbf{u}} \Delta q_k), \quad (6)$$

where $B_r(q_c)$, Δs_k, and Δq_k are the same as in (5) and $\rho(e)$ is an M-estimator. The main difference between (5) and (6) is in the choice of the estimator.

The least-squares (LS) estimator $\rho(e) = e^2$ is used in (5), but the LS estimator is not robust to outliers. Another choice for the estimator is the L_1-norm $\rho(e) = |e|$, which is more robust than LS estimator. However, both have the least possible breakdown point [23]. The breakdown point (BDP) of an estimator is a measure of its resistance to outliers. It refers to the smallest proportion of incorrect samples that the estimator can tolerate before they arbitrarily affect the model fitting [24]. The maximum BDP possible is 50%, because if the number of outliers is larger than the number of inliers, the estimator captures the statistics of the outliers.

The approach based on the influence function [24] characterizes an estimator based on its *influence function*, $\psi(e) = \frac{d\rho}{de}(e)$, and its weight function, $w(e) = \frac{1}{e}\frac{d\rho}{de}(e)$. The data concentrated at the tail of the distribution should not influence the estimation result. In other words, the weight function should assign smaller weights to such data. The LS estimator has a constant weight function and outliers receive the same constant weight as data. M-estimators with a redescending influence function, outperform bounded estimators that are not redescending (for example, L_1-norm) [24], [25].

Several redescending M-estimators have been used in computer vision and robotics literature. Tukey's Biweight (BW) function is popular among the image-based visual tracking and servoing [14]–[17], [26] because of high Gaussian efficiency. The Geman-McClure (GM) estimator has been successfully used in the computer vision community for pattern matching and optical flow estimation [27]. Here, we motivate using the GM robust M-estimator as the robust M-estimator in (6):

$$\rho_{GM}(e; \sigma) = \frac{e^2}{e^2 + \sigma^2},$$

where σ is a measure of scale explained in Section III-B. The influence of the outliers starts to decrease when the residual error is larger than the inflection point of the estimator. For the GM estimator, this happens when $|e| > \sigma/\sqrt{3} \simeq 0.577\sigma$. For BW estimator, the inflection point is larger $|e| > \frac{c}{\sqrt{5}}\sigma \simeq 2.0952\sigma$. Parameter c is a constant of the BW estimator which is set to 4.6851 to achieve 95% Gaussian efficiency [25]. Figure 3 illustrates the discussed estimators and their corresponding influence and weight functions. The BW estimator assigns relatively large weights to residual errors in $[2\sigma, 3\sigma]$. For our visual-motor data set, the outliers tend to be not too far from the inliers and stricter outlier-rejection criteria is desired. The GM estimator does a better job of rejecting the visual-motor outliers. This has been experimentally validated in Section IV (see Figure 5).

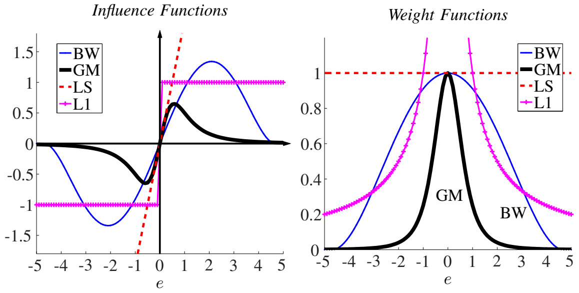

Fig. 3. L_1-norm (L1), Least-squares (LS), Tukey's Biweight (BW), and Geman-McClure (GM) estimators. (Left) Influence functions, (Right) Normalized weight functions, for $\sigma = 1$.

B. Measure of Scale

Scale σ is a parameter that quantifies how the probability distribution is spread. For example, variance is a measure of scale for the normal distribution. Robust measures of scale are needed to estimate the scale parameter in the M-estimator. The Median Absolute Deviation (MAD) is one of the most common estimators of scale, which has the highest possible BDP of 50% and a bounded influence function [24]. Although its Gaussian efficiency is rather low at 37%, it is computationally very efficient [24]. Thus, we use it as a computationally-efficient measure of scale. In essence, MAD estimates the variance of the inlier samples:

$$\sigma = B \operatorname*{med}_{i} \left\{ \left| x_i - \operatorname*{med}_{j}\{x_j\} \right| \right\}, \quad (7)$$

where B is a constant chosen to make MAD consistent with the normal distribution using the cumulative normal distribution function $\Phi(\cdot)$: $B = 1/\Phi^{-1}(3/4) = 1.4826$. Samples $\{x_k\}_{k=1}^{k=K}$ should be scalar because of median, but our visual-motor space is multi-dimensional. We use $x_k = ||\Delta s_k||$, where (s_k, q_k) are the K-nearest neighbors of the query (s_c, q_c) in the joint space.

C. Algorithm

Solving the robust M-estimation problem in (6) can be challenging. A common practice is to solve (6) by iteratively recomputing the weights of a weighted least squares problem [25]. This method is known as the Iteratively Reweighted Least Squares (IRLS) algorithm. The IRLS algorithm is widely used as an efficient implementation of robust M-estimation in many practical nonlinear optimization domains. Examples include robust visual tracking and visual servoing [14]–[17], [26], [27]. Our algorithm can be summarized as follows:

A.1 **Initialize visual-motor memory**: Start by an offline memory initialization for the visual-motor data pairs similar to [9]. This will generate visual-motor memory $\{(s_p, q_p)\}_{p=1}^{P}$.

A.2 **Determine neighbors**: For the visual-motor query (s_c, q_c), determine the neighboring visual-motor pairs in $\{(s_k, q_k)\}_{k=1}^{K}$, which contains the K-nearest neighbors of q_c.

A.3 **Estimate initial scale**: Use MAD as in (7) to find the initial measure of scale σ.

A.4 **Find initial weights:** Initialize weight matrix $\mathbf{W}_0$ according to the norm and scale found in (7). We assign binary weights [16], [23] to the K-nearest neighbors of query point (s_c, q_c) according to:

$$w_k = \left\{ \begin{array}{ll} 1 & : |x_k| \leq 2.5\sigma \\ 0 & : \text{otherwise} \end{array} \right. , \qquad (8)$$

where $\mathbf{W}_0 = \operatorname{diag}[w(e_1) \cdots w(e_K)]$.

A.5 **Estimate the Jacobian:** Given a query point (s_c, q_c), its K-nearest neighbors from the memory $\{(s_k, q_k)\}_{k=1}^{K}$, scale σ, robust estimator $\rho(e;\sigma)$, and the initial weight matrix $\mathbf{W}_0$, JACOBIANESTIRLS is called to estimate the Jacobian. The details of the IRLS algorithm are outlined in Figure 4.

A.6 **Update control signal**: The Jacobian estimated in the previous step is used in (4) to generate the control signal.

A.7 **Update memory**: The new visual-motor pair is added to the memory for later use $P = P + 1$.

A.8 **Return**: Goto step A.2.

```
JACOBIANESTIRLS($(s_c, q_c)$, $\{(s_k, q_k)\}_{k=1}^{K}$, $\sigma$, $\rho(e;\sigma)$, $\mathbf{W}_0$)
    $\mathbf{W} \leftarrow \mathbf{W}_0$, $t \leftarrow 1$, $\widehat{\mathbf{J}}_{\mathbf{u}}(0) \leftarrow \mathbf{0}$
    for $k = 1$ to $K$: $(\Delta s_k, \Delta q_k) \leftarrow (s_c, q_c) - (s_k, q_k)$
    $\Delta S_{[K \times M]} \leftarrow [\Delta s_1 \cdots \Delta s_K]^\top$
    $\Delta Q_{[K \times N]} \leftarrow [\Delta q_1 \cdots \Delta q_K]^\top$
    while $||\widehat{\mathbf{J}}_{\mathbf{u}}(t) - \widehat{\mathbf{J}}_{\mathbf{u}}(t-1)|| > \epsilon$
        $[U, \Sigma, V] \leftarrow SVD(\mathbf{W}\Delta Q)$
        $\widehat{\mathbf{J}}_{\mathbf{u}}(t) \leftarrow [(U\,\Sigma\,V^\top)^\top\,\mathbf{W}\,\Delta S]^\top$
        $[e_1 \cdots e_K]^\top \leftarrow \|\mathbf{W}\Delta Q\,\widehat{\mathbf{J}}_{\mathbf{u}}(t) - \mathbf{W}\Delta S\|$
        for $k = 1$ to $K$: $w(e_k) \leftarrow \frac{1}{e}\frac{\partial}{\partial e}\rho(e;\sigma)\big|_{e=e_k}$
        $\mathbf{W} \leftarrow \operatorname{diag}[w(e_1) \cdots w(e_K)]$
        $t \leftarrow t + 1$
    end
    return $\widehat{\mathbf{J}}_{\mathbf{u}}(\mathbf{t})$
```

Fig. 4. Jacobian Estimation by Iteratively Reweighted Least Squares.

IV. EXPERIMENTS

In this section, we experimentally evaluate the performance of our proposed method (Section III) and compare it with the LLS method [9] (Section II-B) and the actual Jacobian. The results for the quality of Jacobian estimation, as well as the control trajectories with and without outliers, are presented.

A. System Description

An uncalibrated eye-in-hand IBVS system with 4-DOF WAM is considered. Simulations are implemented in MATLAB using the Robotics Toolbox [28] and the Epipolar Geometry Toolbox [29]. The WAM arm runs on RTAI-Linux and controlled with openman [30]. The vision system consists of a Point Grey Grasshopper camera that captures 640×480 MONO8 images at 60 Hz. The Visual Servoing Platform (ViSP) [31] is used for visual tracking. To evaluate the algorithm, we use eight fiducial markers. Therefore, we have 16 visual features (two coordinates for each point in image space) and four joints, *i.e.,* $\widehat{\mathbf{J}}_{\mathbf{u}} \in \mathbb{R}^{16 \times 4}$.

B. Jacobian Estimation Error

Jacobian estimation error is measured by the Frobenius norm of the estimated Jacobian to a reference Jacobian, $\mathbf{J_{ref}}$. In simulations, the exact value of $\mathbf{J_{ref}}$ is known. In real experiments, the reference is found by orthogonal motions around the desired point [8]. This measure makes comparison quantifiable.

The initialization process includes selection of visual features followed by arm motions to store the visual-motor observations into memory. This process takes only a few minutes and is very straightforward. Once a significant number of points are recorded in the memory (approximately 2,000 visual-motor pairs to start), reference Jacobians at 50 random points are estimated using orthogonal motions for ground-truth comparison. Two types of outliers are considered:

1) Type-1 outliers: : Type-1 outliers represent lost features due to occlusion. Lost features are replaced with zero in the visual feature vector. This is done to keep the Jacobian dimension consistent in the the control law in (4). We consider losing only one of the fiducial markers, which associates with two zeros in the visual feature vector. The norm of the corresponding visual-motor outlier is not arbitrarily large, because other visual features are correct.

2) Type-2 outliers: : Type-2 outliers are caused by mis-tracking due to a variety of reasons. The most common is due to confusion of tracking template to nearby templates with a similar appearance. Because of the real-time servoing constraint, descriptive features cannot usually be used. This increases the mis-tracking problem, especially when the robot moves quickly. To model this type of outliers, the corrupted feature is translated by 100 pixels. Other authors have used a similar model [16].

The quality of estimated Jacobian is summarized in Figures 5-6. The performance of the proposed robust estimation with two different M-estimators, Tukey's Biweight (BW) and

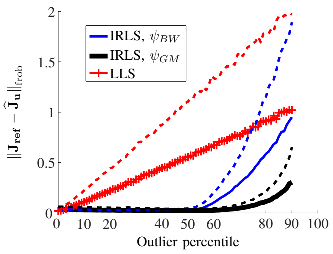

Fig. 5. Simulation results of Jacobian estimation error. The reference Jacobian is found from orthogonal motions. The graph legend corresponds to $K = 200$ and dotted lines to $K = 100$. In practice (see Figure 6), the IRLS method handles outliers up to 40%.

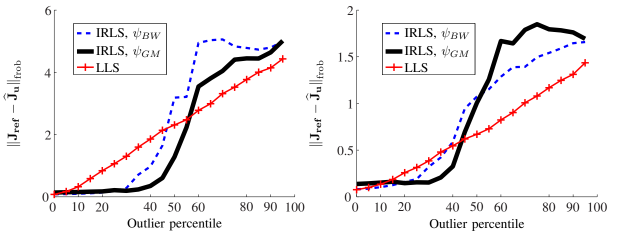

Fig. 6. Real experimental results of Jacobian estimation error for an eye-in-hand WAM. (a) Type-1 outlier, and (b) Type-2 outlier. The GM estimator outperforms others. $K = 200$. (see Section IV-B).

Geman-McClure's (GM), are compared to the least-squares (LLS) performance. In Figure 5, MATLAB simulations are presented. The error norm is averaged over 1,000 random visual-motor points in the workspace for several amounts of outlier/inlier percentage. The outlier/inlier percentage linearly affects the least-squares method. The robust method rejects the outliers up to a certain point, after which the performance breaks down. The GM estimator outperforms the BW estimator in this experiment. This is, in part, due to relatively small norm of the outliers, *i.e.,* the outliers statistically lie between the inflection points of the estimators: $0.577\sigma \leq |e| \leq 2.095\sigma$ (see the discussion at the end of Section III-A). Figure 5 also compares the performance w.r.t. different number of neighbors, K. Two values considered here are $K = 100$ and $K = 200$. With the same size of memory, more neighbors provide a better estimate. However, the neighborhood should satisfy the local-linearity constraint and this number cannot be chosen arbitrarily large. Model selection algorithms such as cross-validation can be used to find the optimal K (see [32], for example).

Figure 6 shows the Jacobian estimation results for the real experimental setup for both types of outliers ($K = 200$). The results of $K = 100$ were similar and have not been overlayed to avoid clutter. The practical breakdown point of the estimators were smaller than their theoretical and simulated values. This is attributed to the non-Gaussian data and the small number of points in the memory. For Type-2 outliers, the BW estimator does not provide a strong robustness. Using the exact same measure of scale (see Section III-B), the GM rejects up to 40% of the outliers with both types of outliers. However, with the increase of outliers, the performance of LLS starts to degrade considerably and robust estimation should be considered.

For the visual servoing experiments in the next sections, we use the GM estimator, because it provides a more robust estimation than BW (see Figures 5-6).

C. Visual Servoing without Outliers

We have used the estimated visual-motor Jacobian using both LLS and IRLS (with GM estimator) in uncalibrated IBVS control law expressed in (4). Figure 7 (a) depicts the end-effector positioning error and Figure 7 (b) shows the norm of image-space error. The goal of this experiment is to show that LLS and IRLS perform similarly when there is an adequate visual-motor memory. Without a dense memory, the K-nearest neighbors of the query point may not conform to the local-linearity constraint. This is due to the highly non-linear visual-motor function. That is, some of the K neighbors will have a relatively large distance to the query point. The robust method downweights such points and reduces their influence on the estimation result. To show this point, we started with a relatively small memory, and let the visual servo drive the arm from an initial position to a desired position. During servoing, new data are added to memory. The LLS method converges to the desired position but at a slower rate than IRLS. At time $t = 40$, the arm is moved back close to the initial point. With more relevant data available in the memory at $t = 40$, both LLS and IRLS converge at the same rate and with similar accuracy. This result is consistent with Figures 5-6.

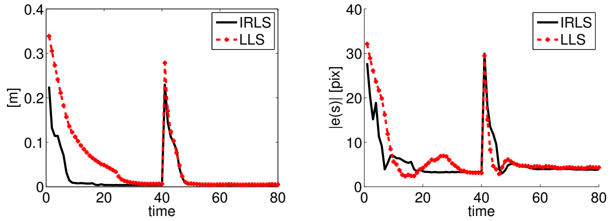

Fig. 7. Error measures for LLS and IRLS *without* outliers. (a) End-effector positioning error. (b) Visual space error. IRLS control converges faster with the initial memory $t \leq 40$. For $t > 40$ the memory is richer and performance is similar.

D. Visual Servoing with Outliers

Finally, we study the effect of outliers on visual servoing performance. We use the same initial starting and desired points as the last section, but add 30% outliers (Type-1) to the data. Figure 8 shows a sample visual servoing performance with outliers introduced at time $t = 60$. The IRLS algorithm manages to estimate a meaningful Jacobian to drive the arm towards the goal. However, LLS gives a wrong estimate, which drives the robot in a wrong direction, where the robot gets stuck in a local minimum. Both LLS and IRLS perform similarly without outliers ($t < 60$). This is in agreement with Figure 6 (a), where a similar type-1 outlier is used. For the purpose of this experiment, GM estimator is used because of its overall robustness to a larger range of outliers. These

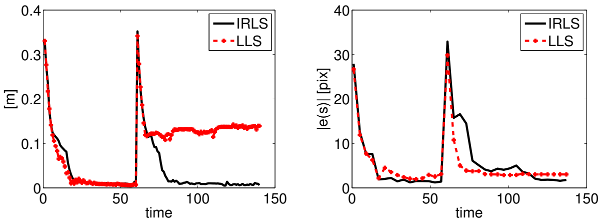

Fig. 8. Error measures for LLS and IRLS with outliers introduced at $t > 60$. (a) End-effector positioning error. (b) Visual space error. The IRLS Jacobian estimates are robust to outliers and drive the arm to the desired point. The LLS estimates are erroneous and drive the robot in the wrong direction. The LLS experiment ends up in a local minimum.

results show that LLS is not robust to outliers, however, IRLS tolerated the outliers and could still reach the desired goal.

V. Conclusions

We proposed a visual-motor Jacobian estimation method using statistically robust M-estimation in the presence of visual-motor outliers. This makes the system less sensitive to outliers as opposed to methods like LLS [9], which show a deteriorating performance in the presence of outliers. In contrast to methods like Broyden rank-one update [7], [8], which cannot exploit the visual-motor memory, our proposed method uses visual-motor memory to gradually increase the quality of the Jacobian estimate.

We compared two M-estimators (Tukey's Biweight and Geman-McClure) with the proposed robust Jacobian estimation against LLS. We used both simulation data and real data from an eye-in-hand 4-DOF WAM robot. The robust methods outperformed LLS in the presence of outliers. Moreover, the Geman-McClure's estimator was superior to the Tukey's Biweight estimator.

Designing a visual-servoing system that uses our robust Jacobian estimation, in conjunction with the robust control signal [16] and/or visual cue integration [10] is left for future work.

References

[1] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE Trans. Robot. Automat.*, vol. 12, no. 5, pp. 651–670, Oct. 1996.

[2] F. Chaumette and S. Hutchinson, "Visual servo control. part I: Basic approaches," *IEEE Robot. Automat. Mag.*, vol. 13, no. 4, pp. 82–90, Dec. 2006.

[3] W. J. Wilson, C. C. W. Hulls, and G. S. Bell, "Relative end-effector control using cartesian position based visual servoing," *IEEE Trans. Robot. Automat.*, vol. 12, no. 5, pp. 684–696, October 1996.

[4] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. Robot. Automat.*, vol. 8, no. 3, pp. 313–326, June 1992.

[5] E. Malis and F. Chaumette, "Theoretical improvements in the stability analysis of a new class of model-free visual servoing methods," *IEEE Trans. Robot. Automat.*, vol. 18, no. 2, pp. 176–186, Apr. 2002.

[6] F. Chaumette and S. Hutchinson, "Visual servo control. part II: Advanced approaches [tutorial]," *IEEE Robot. Automat. Mag.*, vol. 14, no. 1, pp. 109–118, Mar. 2007.

[7] K. Hosoda and M. Asada, "Versatile visual servoing without knowledge of true jacobian," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 1, September 1994, pp. 186–193.

[8] M. Jägersand, O. Fuentes, and R. Nelson, "Experimental evaluation of uncalibrated visual servoing for precision manipulation," in *Proc. IEEE Int. Conf. Robot. Automat.*, vol. 4, April 1997, pp. 2874–2880.

[9] A. M. Farahmand, A. Shademan, and M. Jägersand, "Global visual-motor estimation for uncalibrated visual servoing," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2007, pp. 1969–1974.

[10] D. Kragic and H. I. Christensen, "Cue integration for visual servoing," *IEEE Trans. Robot. Automat.*, vol. 17, no. 1, pp. 18–27, Feb. 2001.

[11] ——, "Robust visual servoing," *International Journal of Robotics Research*, vol. 22, no. 10-11, pp. 923–939, October-November 2003.

[12] P. Preisig and D. Kragic, "Robust statistics for 3D object tracking," in *Proc. IEEE International Conference on Robotics and Automation*, May 2006, pp. 2403–2408.

[13] A. I. Comport, D. Kragic, E. Marchand, and F. Chaumette, "Robust real-time visual tracking: Comparison, theoretical analysis and performance evaluation," in *Proc. IEEE Int. Conf. Robot. Automat.*, Apr. 18–22, 2005, pp. 2841–2846.

[14] T. T. H. Tran and E. Marchand, "Real-time keypoints matching: application to visual servoing," in *Proc. IEEE Int. Conf. Robot. Automat.*, Apr. 10–14, 2007, pp. 3787–3792.

[15] A. I. Comport, E. Marchand, and F. Chaumette, "Kinematic sets for real-time robust articulated object tracking," *Image and Vision Computing*, vol. 25, pp. 374–391, April 2007.

[16] ——, "Statistically robust 2-D visual servoing," *IEEE Trans. Robot.*, vol. 22, no. 2, pp. 415–420, Apr. 2006.

[17] F. Dionnet and E. Marchand, "Robust stereo tracking for space applications," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, Oct. 29 - Nov 2 2007, pp. 3373–3378.

[18] E. Malis, F. Chaumette, and S. Boudet, "2-1/2-d visual servoing," *IEEE Trans. Robot. Automat.*, vol. 15, no. 2, pp. 238–250, Apr. 1999.

[19] C. J. Taylor and J. P. Ostrowski, "Robust vision-based pose control," in *Proc. IEEE Int. Conf. Robot. Automat.*, vol. 3, Apr. 24–28, 2000, pp. 2734–2740.

[20] H. Wang, Y.-H. Liu, and D. Zhou, "Adaptive visual servoing using point and line features with an uncalibrated eye-in-hand camera," *IEEE Trans. Robot.*, vol. 24, no. 4, pp. 843–857, Aug. 2008.

[21] G. Hu, W. MacKunis, N. Gans, W. E. Dixon, J. Chen, A. Behal, and D. Dawson, "Homography-based visual servo control with imperfect camera calibration," *IEEE Trans. Automat. Contr.*, vol. 54, no. 6, pp. 1318–1324, June 2009.

[22] J. T. Lapreste, F. Jurie, M. Dhome, and F. Chaumette, "An efficient method to compute the inverse jacobian matrix in visual servoing," in *Proc. IEEE Int. Conf. Robot. Automat.*, vol. 1, 2004, pp. 727–732.

[23] P. J. Rousseeuw and A. M. Leroy, *Robust regression and outlier detection*. New York, NY: USA: John Wiley and Sons, Inc., 1987.

[24] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel, *Robust Statistics: The Approach Based on Influence Functions*, ser. Wiley series in probability and mathematical statistics. New York, NY: John Wiley and Sons, Inc., 1986.

[25] P. J. Huber, *Robust Statistics*, ser. Wiley Series in Probability and Mathematical Statistics. New York, NY, USA: Wiley-IEEE, 1981.

[26] C. Collewet and F. Chaumette, "Using robust estimation for visual servoing based on dynamic vision," in *Proc. IEEE Int. Conf. Robot. Automat.*, Apr. 10–14, 2007, pp. 2835–2840.

[27] M. Black and A. Jepson, "Eigentracking: Robust matching and tracking of articulated objects using a view-based representation," *Int. J. Comput. Vision*, vol. 36, no. 2, pp. 101–130, 1998.

[28] P. Corke, "A robotics toolbox for MATLAB," *IEEE Robot. Automat. Mag.*, vol. 3, no. 1, pp. 24–32, Mar. 1996.

[29] G. Mariottini and D. Prattichizzo, "EGT: a toolbox for multiple view geometry and visual servoing," *IEEE Robot. Automat. Mag.*, vol. 3, no. 12, December 2005.

[30] S. Leonard. openman: An open source C++ toolbox for control and simulations of manipulators. [Online]. Available: http://sourceforge.net/projects/openman/

[31] E. Marchand, F. Spindler, and F. Chaumette, "ViSP for visual servoing: a generic software platform with a wide class of robot control skills," *IEEE Robot. Automat. Mag.*, vol. 12, no. 4, pp. 40–52, Dec. 2005.

[32] A. M. Farahmand, A. Shademan, M. Jägersand, and C. Szepesvari, "Model-based and model-free reinforcement learning for visual servoing," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 12–17, 2009, pp. 2917–2924.