Proceedings of the 2003 IEEE/RSJ
Intl. Conference on Intelligent Robots and Systems
Las Vegas, Nevada · October 2003

A Visual Servoing Control Law that is Robust to Image Outliers

Andrew Comport, Muriel Pressigout, Éric Marchand, François Chaumette
IRISA - INRIA Rennes
Campus de Beaulieu, 35042 Rennes Cedex, France
E-Mail: Firstname.Name@irisa.fr

***Abstract*— A fundamental step towards broadening the use of real world image-based visual servoing is to deal with the important issues of reliability and robustness. In order to address this issue, a closed loop control law is proposed that simultaneously accomplishes a visual servoing task and is robust to a general class of external errors. This generality allows concurrent consideration of a wide range of errors including: noise from image feature extraction, small scale errors in the tracking and even large scale errors in the matching between current and desired features. This is achieved with the application of widely accepted statistical techniques of robust M-estimation. The M-estimator is integrated by an iteratively re-weighted method. The Median Absolute Deviation is used as an estimate of the standard deviation of the inlier data and is compared with other methods. This combination is advantageous because of its high efficiency, high breakdown point and desirable influence functions. The robustness and stability of the control law is shown to be dependent on a subsequent measure of position uncertainty. Furthermore the convergence criteria of the control law are investigated. Experimental results are presented which demonstrate visual servoing tasks which resist severe outlier contamination.**

I. Introduction

Visual servoing is targeted at controlling the movement of robotic systems by exploiting image sensor information. A general task being to move an end-effector to a certain pose with respect to particular objects or features in the image. This is known to be a very efficient method for positioning tasks [7], however, its efficiency is subject to varying degrees of error. The efficiency of visual servoing relies on correspondences between the position of tracked visual features in the current image and their desired positions in the desired image. These correspondences are typically exploited in the form of a image error to be minimized. If these correspondences contain errors then visual servoing usually fails or convergences upon an imprecise position.

Traditionally, robustness of the control law has been defined as "stability results which remain true in the presence of *modeling errors* or certain classes of *disturbance*" [12]. Here two typical solutions emerge. The first is to create a more accurate model of the system and the other is to treat the different classes of disturbances. In the first solution, it is true that accurately modeling intrinsic parameters reduces modeling error and improves results. In visual servoing this has lead to modeling of the camera in terms of perspective projection and estimating the depth parameter online [9], [3], [14]. Other sources of modeling error include those introduced by local detection and matching of features between the first and desired images. Overcoming this class of error is often achieved by improving the quality of tracking algorithms [15] and feature selection methods [11]. These models are fundamentally important, however, in visual servoing little work has been done on rejecting the external disturbances by using well founded robust statistical techniques and visual servoing.

The solutions mentioned in the previous paragraph are partial solutions to categorized error by further modeling, however, a real time image sequence represents an almost infinite source of information which is extremely difficult to model exhaustively by analytical development. This includes dynamic problems such as moving objects, multiple occlusions, changes in illumination due to varying light sources, different visual medium such as water and air, etc... It is clear that handling all the potential sources of error by analytical classification is a very complex and difficult task. In this paper the problem of *statistically robust* visual servoing is implemented directly at a transient control law level. In order to represent all the possible external sources of error the correspondences may contain, a statistical measure of position uncertainty is sought.

In related computer vision and statistics literature many different approaches exist to treat external sources of error. Amongst the robust outlier rejection algorithms in the literature, methods in computer vision have included the Hough Transform and RANSAC [2] and statistical methods such as LMedS and M-estimators [6]. Up to present most approaches focus only on a single geometric error function and reject any *outliers* which do not correspond to this definition. The reader is referred to [13] for a review of different robust techniques applied to computer vision.

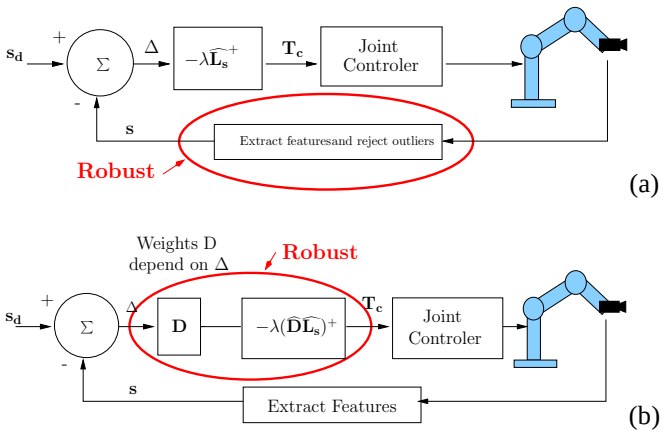


Fig. 1. (a) Traditional outlier rejection (b) New proposed control law (see Section II-B for details)

In the visual servoing literature, Kragic and Christensen [8] have previously proposed an approach to robustness by using a voting and consensus technique. This approach provides a robust input estimate (cue) to a control loop and as such treats outlier rejection, in an image processing step, prior to the visual servoing

task (see Figure I). Furthermore, this type of method requires the tuning of parameters to deal with the complexity of the system. In this paper robust M-estimators are employed because they give a solid statistical basis for detailed analysis and have even been considered to be a unifying banner for these estimation techniques [10]. The estimation of the scale using the Median Absolute Deviation (MAD) means that no tuning is required. Furthermore, formulation in terms of an Iteratively Re-weighted Least Square (IRLS) allows simple integration directly into the visual control law. Hence, the accuracy, efficiency and stability in the face of outlier data are easily demonstrated. This can be explained intuitively by the fact that the outliers are not rejected until the control law has converged sufficiently to be sure that an outlier actually exists at a certain position.

Following an introduction to the method, a robust control scheme is proposed in Section II-B. This is achieved by introducing a weight matrix in the error minimization. We present, in Section II-C, how the use of the M-estimators allows computation of the weights which reflect a confidence in each feature in the image. Experimental results are presented in Section III.

II. Robust Visual Servoing

A. Overview and motivations

The goal of classical visual servoing [1], [7] is essentially to minimize the error Δ between a set of image features $\mathbf{s}(\mathbf{r})$, that depends of the camera pose $\mathbf{r}$, and a set of desired image features $\mathbf{s_d}$:

$$\Delta = \mathbf{s}(\mathbf{r}) - \mathbf{s_d}. \tag{1}$$

The camera then has to reach the pose $\mathbf{r_d}$ that minimizes this error.

However, as stated in the previous section, considering that $\mathbf{s}(\mathbf{r})$ is computed (from the image) with a sufficient accuracy is an important assumption. The control law that performs Δ minimization is usually handled using a least square approach [1], [7]. However when the data contains outliers, a robust minimization is required. M-estimators can be considered as a more general form of Maximum Likelihood Estimators (MLE) [6] because they permit the use of different minimization functions not necessarily corresponding to normally distributed data. Many functions have been presented in the literature which allow uncertain measures to be less likely considered and in some cases completely rejected. In the following subsections ρ is the objective function considered. The metric function to be minimized is modified to reduce the sensitivity to outliers. The robust optimization problem is then given by:

$$\Delta_{\mathcal{R}} = \rho\Big(\mathbf{s}(\mathbf{r}) - \mathbf{s_d}\Big) \tag{2}$$

where $\rho(u)$ is a robust function [6] that grows sub-quadratically and is monotonically nondecreasing with increasing $|u|$.

To embed a robust minimization in visual servoing, a modification of the control law is required to allow outlier rejection. The new control law is given in the next subsection while the weight computation method is presented in Section II-C.

B. Robust Control Law

The objective of the control scheme is to minimize the objective function given in equation (2). This new objective is incorporated into the control law in the form of a weight, which is given to specify a confidence in each feature location. Thus, the task function is given by:

$$\mathbf{e} = \mathbf{CD}(\mathbf{s}(\mathbf{r}) - \mathbf{s_d}), \tag{3}$$

where

- matrix $\mathbf{C}$ is the usual combination matrix of size $m \times k$ where k is the number of features and m the number of controlled robot degrees of freedom (6 to reach a unique desired position),
- $\mathbf{D}$ is a diagonal weighting matrix given by

$$\mathbf{D} = \mathbf{diag}(\mathbf{w_1}, \ldots, \mathbf{w_k})$$

The computation of weights w_i are described in Section II-C.

If $\mathbf{C}$ and $\mathbf{D}$ were constant, the derivative of equation (3) would be given by:

$$\dot{\mathbf{e}} = \frac{\partial \mathbf{e}}{\partial \mathbf{s}} \frac{\partial \mathbf{s}}{\partial \mathbf{r}} \frac{d\mathbf{r}}{dt} = \mathbf{CDL_s T_c}, \tag{4}$$

where $\mathbf{T_c}$ is the camera velocity and $\mathbf{L_s}$ is called the image Jacobian [7] or interaction matrix [1] related to $\mathbf{s}$. This matrix depends on the value of the image features $\mathbf{s}$ and their corresponding depth Z in the scene. If we specify an exponential decrease of the error $\mathbf{e}$:

$$\dot{\mathbf{e}} = -\lambda \mathbf{e}, \tag{5}$$

where λ is a positive scalar, the following control law is obtained from equation (4):

$$\mathbf{T_c} = -\lambda(\mathbf{C}\widehat{\mathbf{D}}\widehat{\mathbf{L_s}})^{-1}\mathbf{e}, \tag{6}$$

where $\widehat{\mathbf{L_s}}$ is a model or an approximation of the real matrix $\mathbf{L_s}$ and $\widehat{\mathbf{D}}$ a chosen model for $\mathbf{D}$.

To simplify the control law, $\mathbf{C}$ can be chosen to be the pseudo inverse $(\widehat{\mathbf{D}}\widehat{\mathbf{L_s}})^+$ of $\widehat{\mathbf{D}}\widehat{\mathbf{L_s}}$. This gives $\mathbf{C}\widehat{\mathbf{D}}\widehat{\mathbf{L_s}} = (\widehat{\mathbf{D}}\widehat{\mathbf{L_s}})^+\widehat{\mathbf{D}}\widehat{\mathbf{L_s}} = \mathbf{I_m}$. Finally giving (see Figure I):

$$\mathbf{T_c} = -\lambda(\widehat{\mathbf{D}}\widehat{\mathbf{L_s}})^+\mathbf{D}\big(\mathbf{s}(\mathbf{r}) - \mathbf{s_d}\big), \tag{7}$$

If $\widehat{\mathbf{D}}$ and $\widehat{\mathbf{L_s}}$ were constant, a sufficient criteria to ensure global asymptotic stability of the system would be given by [1], [12]:

$$(\widehat{\mathbf{D}}\widehat{\mathbf{L_s}})^+\mathbf{DL_s} > \mathbf{0} \tag{8}$$

As usual, in image-based visual servoing, it is impossible to demonstrate the global stability. It is, however, possible to obtain the local stability for two cases of $\widehat{\mathbf{L_s}}$ and $\widehat{\mathbf{D}}$:

- the first case is to use the current value of the weights, an estimate of the depth at each iteration (if available) and the current feature:

$$(\widehat{\mathbf{D}}\widehat{\mathbf{L_s}})^+ = \big[\mathbf{DL_s}(\mathbf{s}, \hat{\mathbf{Z}})\big]^+ \tag{9}$$

This choice allows the system to follow as closely as possible the intended behavior ($\dot{\mathbf{e}} = -\lambda\mathbf{e}$). However, even when condition (8) is satisfied, only local stability can be demonstrated since $\mathbf{D}$ and $\mathbf{L_s}$ are not constant (refer to (4) that has been used to derive (8)).

- In the second case a constant Jacobian is considered using the desired depth $\mathbf{Z_d}$, the desired value of the features $\mathbf{s_d}$ and the first value of the weighting matrix $\widehat{\mathbf{D}} = \mathbf{I}_k$.

$$(\widehat{\mathbf{D}}\widehat{\mathbf{L_s}})^+ = \mathbf{L_s}(\mathbf{s_d}, \mathbf{Z_d})^+, \tag{10}$$

The main advantage of this is that the depth does not have to be computed at each iteration. Furthermore this leads to a simpler control law:

$$\mathbf{T_c} = -\lambda\mathbf{L_s}(\mathbf{s_d}, \mathbf{Z_d})^+\mathbf{e} = -\lambda\mathbf{L_s}(\mathbf{s_d}, \mathbf{Z_d})^+\mathbf{D}\big(\mathbf{s} - \mathbf{s_d}\big) \tag{11}$$

and a simpler convergence sufficient condition:

$$\mathbf{L_s}(\mathbf{s_d}, \mathbf{Z_d})^{+}\mathbf{DL_s} > \mathbf{0}. \quad (12)$$

Note also that, even if model (10) is constant, the evolution of the weights during the realization of the control law are taken into account through the computation of $\mathbf{e}$, as in (11). Furthermore, the weights $\mathbf{w_i}(0)$ could be computed instead of choosing them to be equal to 1, however, these initial weights may be equally incorrect. Finally, only the local stability of the system can be demonstrated since equation (12) is only satisfied around $\mathbf{s_d}$.

Of course it is also necessary to ensure that a sufficient number of features will not be rejected so that $\mathbf{DL_s}$ is always of full rank (6 to control the 6 dof of the robot).

In the results presented in Section II-C, we have chosen the second option presented with a constant Jacobian given by (10). Experimental experience has shown, however, that even if the stability can only be theoretically demonstrated in a neighborhood of the desired position, this neighborhood is quite large.

C. Computing the weights

The weights w_i, which represent the different elements of the $\mathbf{D}$ matrix and reflect the confidence of each feature, are usually given by [6]:

$$w_i = \frac{\psi(\delta_i/\sigma)}{\delta_i/\sigma} \quad (13)$$

where $\psi\Big(\delta_i/\sigma\Big) = \frac{\partial\rho\left(\delta_i/\sigma\right)}{\partial \mathbf{r}}$ (ψ is the M-estimate and is also called the influence function) and δ_i is the normalized residue given by $\delta_i = \Delta_i - Med(\Delta)$ (where $Med(\Delta)$ corresponds to the median value taken across all the residues).

Of the various influence functions that exist in the literature we consider Huber's monotone function and Tukey's hard re-descending function [6](see Figure 2).

Huber's function asymptotically reduces the influence of an outlier toward zero. The Huber estimator assumes that all values within the bounds of 95% of the data are 100% correct and gradually reduces the probability of features outside this region. Huber influence function is given by:

$$\psi(u) = \begin{cases} u & , \text{ if } |u| \leq a \\ a\frac{u}{|u|} & , \text{ if } |u| > a \end{cases} \quad (14)$$

where the proportionality factor for Huber's function is $a = 1.2107$ which represents 95% efficiency in the case of Gaussian Noise [5].

Tukey's function completely rejects outliers and gives them a zero weight. This is of interest in visual servoing so that detected outliers have no effect on the robot motion. Its corresponding influence function is given by:

$$\psi(u) = \begin{cases} u(b^2 - u^2)^2 & , \text{ if } |u| \leq b \\ 0 & , \text{else} \end{cases} \quad (15)$$

where the proportionality factor for Tukey's function is $b = 4.6851$ which represents 95% efficiency in the case of Gaussian Noise [5].

Figure 2 permits easy comparison of the different functions and their convergence characteristics. Both Huber's and Tukey's functions guarantee unique solutions and give a minimizable partial derivative with respect to the pose parameters. It must be noted that the previous convergence criteria, given in Section II-B, were based on the assumption that the weights $\mathbf{D}$ correctly represented the outliers. Now that the weights have been given by equation (13), it is possible to say that "if the geometric configuration of the visual servoing task is such that it will converge, then so does the modified robust control law". In other words the control law converges with outliers. The scale σ which appears

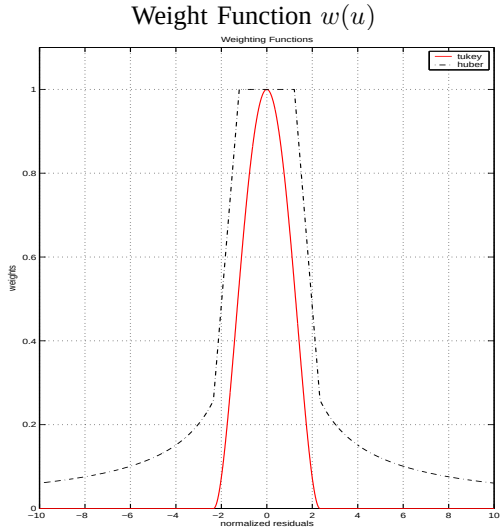


Fig. 2. Simultaneous plot of different weight function of the M-estimators for $\sigma = 0.5$

in (13) is a robust estimate of the standard deviation of the good data and is at the heart of the robustness of the function. In visual servoing this estimate of the scale can vary dramatically during convergence. For voting methods and traditional M-estimators, scale has usually been treated as a tuning variable which can be chosen manually for a specific application. Alternatively, a robust statistic can be used to calculate it. One robust statistic is the Median Absolute Deviation (MAD), given by:

$$\widehat{\sigma} = \frac{1}{\Phi^{-1}(0.75)} Med_i(|\delta_i - Med_j(\delta_j)|). \quad (16)$$

where $\Phi(.)$ is the cumulative normal distribution function and $\frac{1}{\Phi^{-1}(0.75)} = 1.48$ represents one standard deviation of the normal distribution. To date, a convergence proof for non-linear regression using the MAD only exists if it is calculated once as an auxiliary scale estimate due to the median's lack of asymptotic properties [5]. However, although convergence has not yet been proved due to discontinuities introduced by the median, experiments show that recomputing the MAD at each iteration gives better results (see next section).

III. Experimental results

The complete implementation of the robust visual servoing task, including tracking and control, was carried out on an experimental test-bed involving a CCD camera mounted on the end effector of a six d.o.f robot. We have considered positioning tasks. From an initial position, the robot has to reach a desired position expressed as a desired position of the object in the image.

a) Visual features and weights computation.: In these experiments visual features are given as a set of point coordinates extracted from the image. If n points are considered, $\mathbf{s}$ is a vector defined as $\mathbf{s} = (x_1, y_1, x_2, y_2, \ldots, x_n, y_n)$ where (x_i, y_i) are the

coordinates of the $i-th$ point. Interaction matrix $\mathbf{L_s}$ is a $2n\times 6$ matrix given by $\mathbf{L_s} = (\mathbf{L_{s1}}, \ldots, \mathbf{L_{sn}})$ with:

$$\mathbf{L_{s}}_i = \begin{pmatrix} -\frac{1}{Z_i} & 0 & \frac{x_i}{Z_i} & x_i y_i & -(1+x_i^2) & y_i \\ 0 & -\frac{1}{Z_i} & \frac{y_i}{Z_i} & 1+y_i^2 & -x_i y_i & -x_i \end{pmatrix}$$

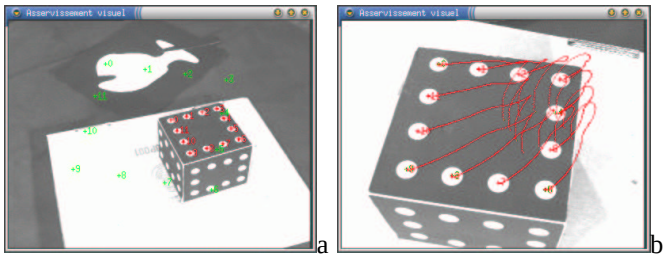
Weights are computed using equation (13) where

$$\forall k = 1 \ldots n, \quad \begin{cases} \Delta_{2k} = x_k - x_{k_d} \\ \Delta_{2k+1} = y_k - y_{k_d} \end{cases}$$

However since weight w_{2k} and w_{2k+1} reflect the confidence we have in the same point, we define elements of the weights matrix $\mathbf{D}$ as

$$\mathbf{D}_{2k,2k} = \mathbf{D}_{2k+1,2k+1} = \min(w_{2k}, w_{2k+1}).$$

b) Experiments with dots: In the first experiment some simple visual servoing experiments are considered that are based on the tracking of a pattern made with twelve white dots. Tracking such a simple pattern allows to validate the efficiency of the new control law. Indeed due to this simplicity, if no noise is artificially introduced in the point matching or in the tracking, a "control-case" is then available.

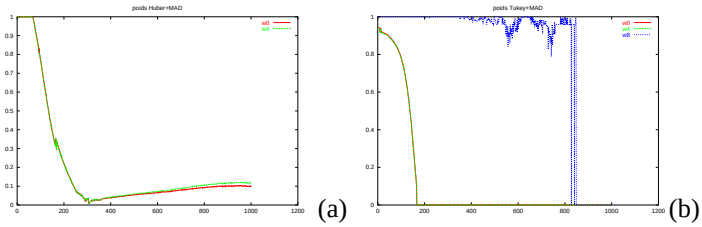


a b

Fig. 3. Classical visual servoing : initial and desired images (red points corresponds to the points trajectories in the image while green crosses are the desired points positions).

In the first trial (reported in Figure 4) an important error was introduced into the extracted coordinates of two points (two points 0 and 2 on Figure 4 were voluntarily inverted). With a classical visual servoing control law the final position was found to be very different from the expected one (see Table I). Camera trajectory and points trajectories in the image space are not satisfactory (the camera oscillates and after 2500 iterations the control law has not converged yet). A robust control law was then applied which considered the Huber and Tukey M-estimators. Huber detects the outliers points but as can be seen on Figure 2, the values of weights w_i never reach zero. All the measures are then taken into account (even lightly) in the control law which implies an error in the positioning task (see Figure 5a and Table I, lines 5). At least, with respect to the previous experiment (no robust) the control law converges as shown on Figure 4 where the error in the image is very small.

We then consider the Tukey M-estimator without and with the computation of the standard deviation of the noise (MAD). In both cases, the points are detected as outliers and are detected faster than with the Huber estimator (and even faster when the MAD σ is computed at each iteration). It is important to note that in this experiment the weights w_i become null which implies a far better positioning accuracy. The error is even smaller than with the classical visual servoing approach (Table I, line 3) since a third point is classified as an outlier (point 4 on Figure 4). This result shows the efficiency of the new robust control law and



(a) (b)

Fig. 5. Computed weights for outliers points (a) with Huber (b) with Tukey (Points 0, 2, and 5 are considered as outliers: only points 0 and 2 are voluntarily inverted.)

the interest of the Tukey M-estimator with respect to the other estimators.

Figure 6 shows the trajectory of the camera optical center for the different experiments. The "reference" trajectory (i.e., classical visual servoing with no noise – points are not inverted –) is displayed in red. The classical visual servoing control law with noise is displayed in green (this trajectory is very noisy, unstable and does not converge toward an equilibrium. The new control law with the Huber estimator in blue (the camera trajectory is smooth and converges, though not exactly toward the desired position). Finally the trajectory in pink corresponds to the control law with the Tukey estimator. This trajectory is obviously different from the "reference" one (in red) since outliers are not detected at the first iteration. However the final camera position is the same (see also Table I which summarizes the positioning results).

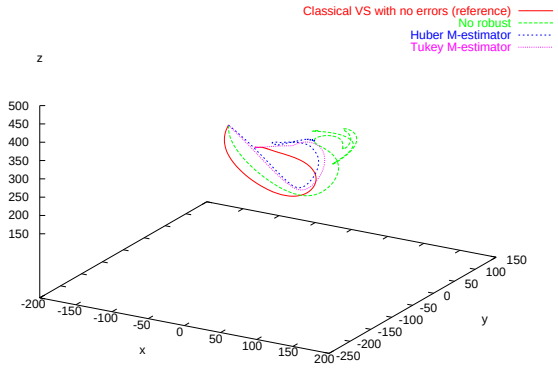


Fig. 6. Camera trajectory for important matching errors.

In the second experiment a small scale error is added to the extracted position of four dots. This is done via a partial occlusion of the acquired images which in turn adds a small error onto the center of gravity of several points (see figure 7). Note that the desired image is captured with no occlusion. Even though the error is quite small, the repositioning errors are significant using classical visual servoing (see Figure 7.b with respect to Figure 3.b). With the robust control (Tukey) and if the standard deviation of the noise measure (MAD) is not recomputed at each iteration, a similar result is obtained since the outliers are very difficult to detect from the first computation of δ where $\delta = (\delta_1, \ldots \delta_n)$ from equation (13). However, if the MAD σ is recomputed, after several iterations (that is when δ has decreased and therefore when the effects of the outliers

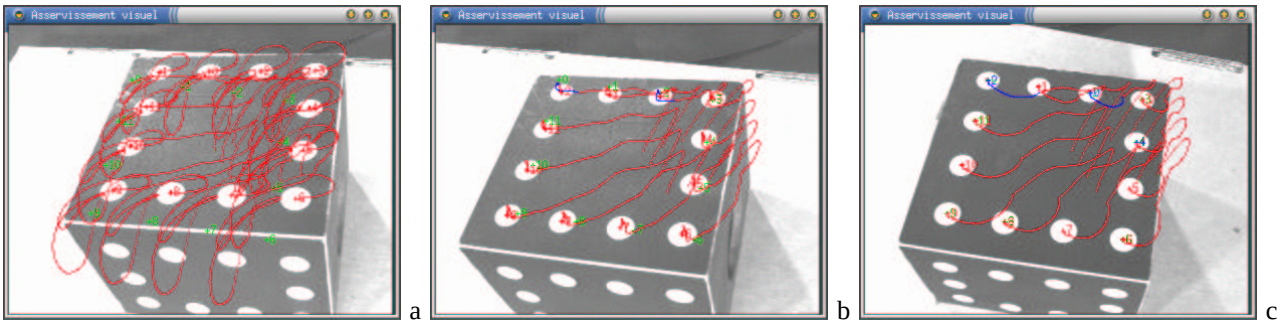


Fig. 4. Final image acquired by the camera with large errors in point extraction (points 0 and 2 are voluntarily inverted in the matching process): (a) no robust control law (no convergence) (b) with the Huber M-estimator (c) with the Tukey M-estimator

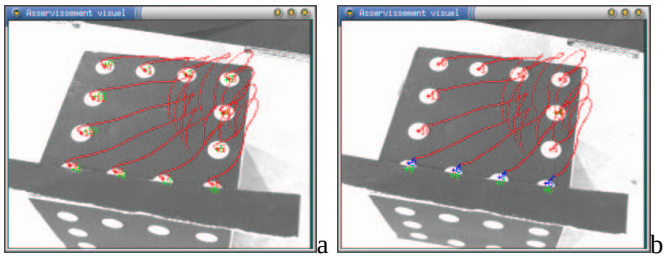


Fig. 7. Small errors on many points (a "row" of points is partially hidden): (a) Initial image, (b) Final image with robust estimation (Tukey + MAD computation at each iteration)

are more visible) the outliers are detected and the repositioning task is achieved with a far better accuracy. As for the previous experiment camera trajectory are proposed on Figure 8 and positioning accuracy is summarized in Table I.

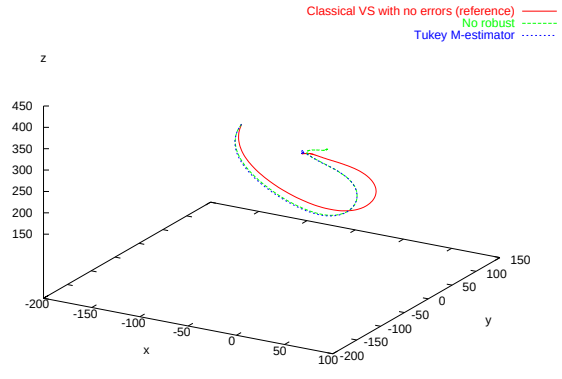


Fig. 8. Camera trajectory for small errors in dots extraction.

c) Experiments with SSD trackers.: In the following experiment we have considered a visual servoing experiment on far more complex images. We defined the position to reach with a reference image (see Figure 9b). Points of interest are extracted (using the Harris detector [4]) and are matched with similar points extracted from the image acquired from the initial camera location. This matching process is done using the Image-Matching software [16]. Tracking during the visual servoing experiment is based on a classical SSD algorithm. In addition to this, the camera is replaced by a simple 30 euro web-cam that provides poor quality images. The feature tracking is thus not very reliable and poor images quality induces errors into a classical visual servoing approach.

Firstly, it can be noted that with the use of a classical control law and due to excessive miss-tracking, the camera was not able to reach the desired position. Figure 9c shows the difference between the desired image (Figure 9b) and the last one acquired by the camera (Figure 9d). In Figure 9d, 9f and 9h red crosses are the initial points location, blue crosses are their desired locations while the green crosses are the final points location. Point trajectories are in red (60 points are tracked). Next the robust control law was applied using both the Huber and Tukey M-estimators. In both cases the MAD was computed at each iteration. Both M-estimators provide similar results, although Tukey's function is preferential since it allows complete rejection of outliers ($w_i = 0$) which is not the case with the Huber estimator. In both cases the desired position was obtained with good accuracy (less that 3 cm in translation) even with very poor experimental conditions. In general the difference between the desired image (Figure 9b) and the last one acquired by the camera is very small (see Figure 9e and Figure 9g).

IV. Conclusion

A novel visual servoing approach has been considered that rejects errors in feature extraction, tracking and matching at the transient control law level. To achieve this goal, a robust M-estimation was integrated directly via an iteratively re-weighted least squares method. Previous visual servoing methods have only considered outlier rejection in the image processing step. Convergence conditions are considered and experimental results show the efficiency of the approach for a repositioning task on both a case-study example and on real images. In both cases a great improvement in the repositioning accuracy has been observed.

This work has only considered image-based tasks defined in the image space. Another limitation in this work is that only point features have been considered. Future methods will look at considering robustness for position-based control of a manipulator and mixtures of this with image-based visual servoing. More complex features will be considered to further improve robustness.

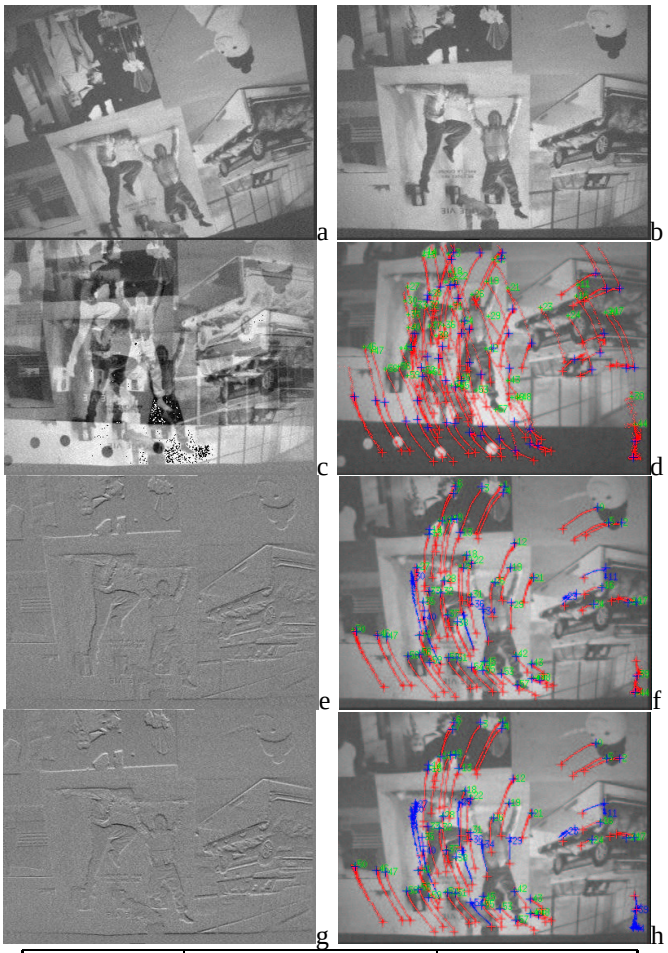
V. References

[1] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation*, 8(3):313–326, June 1992.

	Experiments	T_x	T_y	T_z	R_x	R_y	R_z	$\|\mathbf{T}\|$
1	Initial position	-155.2	124.4	218.2	30.56	46.84	-0.98	
2	Desired position	61.1	-185.2	441.5	0.0	29.45	0	
3	Classical VS no error	-0.9	-0.3	0.6	-0.29	-0.04	0.16	1.12
4	Important errors no robust	47.9	69.1	19.3	6.23	10.54	-13.73	86.26 (E)
5	Important errors Huber	56.1	45.2	10.4	3.42	8.25	-12.3	72.79
6	Important errors Tukey no MAD	-0.2	-0.3	0.6	-0.27	0	0.01	0.7
7	Important errors Tukey with MAD	-0.3	0.2	0.5	-0.27	0.02	0.02	0.6
8	Small errors no robust	18	17.6	7.4	0.9	3.72	-3.97	26.23
9	Small errors Tukey MAD	-1.2	0.7	0.4	-0.29	-0.05	0.25	1.44

TABLE I

Positioning accuracy summary: each line display the error (in mm and degree) between the desired and final camera position.



final position	T_x	T_y	T_z	R_x	R_y	R_z
Desired	-131.2	-17.6	259.2	-0.1	19.2	-0.5
Without robust	-257.6	-356.7	295.8	-0.9	-5.1	8.0
Huber + MAD	-133.5	-13.7	258.4	-0.1	19.4	-0.2
Tukey + MAD	-137.3	-16.4	258.6	-0.1	19.2	0.1

i

Fig. 9. Visual servoing based on the tracking of points of interest: (a) initial image, (b) final image, (c-d) classical visual servoing control law (difference image and point trajectory on the final image), (e-f) robust visual servoing using the Huber M-estimator, (g-h) robust visual servoing using the Tukey M-estimator, (i) repositioning errors (mm and degrees)

[2] N. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography. *Communication of the ACM*, 24(6):381–395, June 1981.

[3] E. Grosso, G. Metta, A. Oddera, and G. Sandini. Robust visual servoing in 3-d reaching tasks. *IEEE Trans. on Robotics and Automation*, 12:732–742, October 1996.

[4] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Conference*, pages 189–192, 1988.

[5] P.-W. Holland and R.-E. Welsch. Robust regression using iteratively reweighted least-squares. *Comm. Statist. Theory Methods*, A6:813–827, 1977.

[6] P.-J. Huber. *Robust Statistics*. Wiler, New York, 1981.

[7] S. Hutchinson, G. Hager, and P. Corke. A tutorial on visual servo control. *IEEE Trans. on Robotics and Automation*, 12(5):651–670, October 1996.

[8] D. Kragic, H.I. Christensen. Cue integration for visual servoing. *IEEE Trans. on Robotics and Automation*, 17(1):18–27, February 2001.

[9] S.-J. Maybank and O. Faugeras. A theory of self calibration of a moving camera. *Int. Journal of Computer Vision*, 8(1):123–152, 1992.

[10] P. Meer, C. V. Stewart, and D. E. Tyler. Robust computer vision: An interdisciplinary challenge. *Computer Vision and Image Understanding*, 78(1):1–7, 2000.

[11] N. P. Papanikolopoulos and P. K Khosla. Selection of features and evaluation of visual measurements for 3-d robotic visual tracking. *Int. Symp. on Intelligent Control.*, pages 320–325, August 1993.

[12] C. Samson, M. Le Borgne, and B. Espiau. *Robot Control: the Task Function Approach*. Clarendon Press, Oxford, 1991.

[13] C.-V. Stewart. Robust parameter estimation in computer vision. *SIAM Review*, 41(3):513–537, September 1999.

[14] C. J. Taylor, J. P. Ostrowski, and S. H. Jung. Robust visual servoing based on relative orientation. *IEEE Int. Conf on Computer Vision and Pattern Recognition*, pages 574–580, 1999.

[15] T. Tommasini, A. Fusiello, E. Trucco, and V. Roberto. Making good features track better. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 178–183, Santa Barbara, June 1998.

[16] Z. Zhang, R. Deriche, O. Faugeras, and Q.-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78:87–119, October 1995.