

# Bucketing Techniques in Robust Regression for Computer Vision

Ariel Choukroun<sup>1</sup> and Vincent Charvillat<sup>1</sup>

IRIT-ENSEEIHU UMR CNRS 5505  
2 rue Camichel BP 7122 31071 Toulouse Cedex 7  
{ariel.choukroun,vincent.charvillat}@enseeiht.fr

**Abstract.** Robust parameter estimation methods have become very popular in the computer vision community. Nevertheless, both optimization models and resolution algorithms coming from robust statistics must be adapted to correctly tackle the specificities of visual data. Among these adapted techniques, computer-vision researchers frequently use bucket-based partitions of the data (bucketing techniques). This work points out the key ideas and features of bucketing techniques. A new stochastic sampling scheme is proposed and defended. We also try to answer several questions, which are generally –and perhaps voluntarily– bypassed : “does the bucketing strategy influence the regression process ?” ; “how should the data be split into buckets to get the best fits both numerically and physically ?” ...

## 1 Introduction

Solving computer vision problems often requires a fundamental step of parameter estimation. The present work focuses on resolution techniques for regression problems. We represent here a regression problem as a black box :

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} \longrightarrow \beta = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} \longrightarrow y = f(x, \beta) = \begin{pmatrix} f(x, \beta)_1 \\ \vdots \\ f(x, \beta)_q \end{pmatrix}$$

$f$  represents a physical system which takes a vector  $x \in \mathbb{R}^m$  for input and returns a vector  $y \in \mathbb{R}^q$ , depending on the parameter vector  $\beta \in \mathbb{R}^p$ .

The main task is to estimate  $\hat{\beta} \in \mathbb{R}^p$  so that the model can predict as well as possible a set of output measures  $\{\tilde{y}_i\}_{i=1..n}$ , resulting from a set of stimuli  $\{\tilde{x}_i\}_{i=1..n}$ .

A common way to solve this is to minimize a least squares criterion. We associate to each measure a residual distance  $r_i^2 = r_i^2(\beta) = \|\tilde{y}_i - f(\tilde{x}_i, \beta)\|^2$ .

The sum of these residuals is then minimized in order to estimate optimal parameters in the Ordinary Least Squares (OLS) sense:  $\text{Min}_{\beta \in \mathbb{R}^p} \sum_{i=1}^n r_i^2(\beta)$

However, the OLS scheme cannot be used in cases where gross errors contaminate input or output measures. That is why Computer Vision researchers widely use so-called ”robust” methods.

This paper is constructed as follows. First, several robust estimators and algorithms are presented. Then, we identify the main ideas that allowed Vision researchers to adapt general methods coming from robust statistics. Finally, we propose a new stochastic sampling scheme and defend it.

## 2 Robust estimation

Classical robust estimators, as  $LMS^1$ ,  $LTS^2$ ,  $TLS$  or  $ODR$  are described in [12, 1, 6, 5, 8] and have been widely used [15, 18]. In practice, these estimators cannot be computed exhaustively. Since the median function is not differentiable, the optimization problem (for instance in the LMS case) cannot be solved exactly (excepted the LMS-based line fitting case in  $\mathbb{R}^2$ ).

In the general case a random sampling technique is required : the key idea, also found in RANSAC [3] or PROGRESS [11], is to randomly select a number  $n'$  of measures to provide an hypothesized fit. The general algorithm presented in figure 1 exploits these candidates. If we call  $n_0$  the *minimal* number of measures which allows the estimation, choosing  $n' = n_0$  (minimal subsets principle) avoids (with a great probability) being systematically contaminated with outliers.

1. Choose sample data size  $n'$  ( $n_0 \leq n' \leq n$ ).  
In the minimal subsets case,  $n' = n_0$ .
2. **For**  $j = 1..N$  **Do**
  - Build a sample of size  $n'$  among the  $(\tilde{x}_i, \tilde{y}_i)_{i=1..n}$ .
  - $\beta^j \leftarrow$  solution for the OLS problem built with this sample.
  - Evaluate the problem criterion (for example median of residuals).
3. **End For**
4. Keep the  $\beta^j$  associated with the best criterion.

**Fig. 1.** Classical random sampling algorithm for LTS or LMS criterions minimization

The choice of a  $N$  big enough to make this stochastic algorithm efficient has been studied [12].

## 3 Works specific to Vision

### 3.1 Strong hypothesis due to visual data nature.

We point out here the inefficiency of the classical algorithms for Computer Vision data and problems. Since sampling is random (following the minimum subsets principle), these samples risk being spatially (that means visually) badly arranged.

<sup>1</sup>  $\hat{\beta} = \operatorname{argmin}_{\beta \in \mathbb{R}^p} \operatorname{Median}_{i=1..n} r_i^2$

<sup>2</sup>  $\hat{\beta} = \operatorname{argmin}_{\beta \in \mathbb{R}^p} \sum_{i=1}^{\lfloor n/2 \rfloor} r_{\sigma(i)}^2$ , where  $\sigma$  is the permutation linked with the sorted residuals.

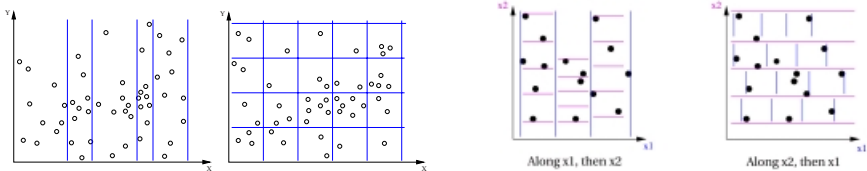
**Avoid a physically unfounded search** Physically, the fits we are looking for have to be valid on the whole working space. For example, a strong calibration has to be homogeneous for the entire working environment. It would be pointless to obtain a perfect camera model from a small 3D calibration target which would be valid only on a small part of the image. Hence an hypothesized fit  $\beta^j$  coming from a random sample (step 2 of algorithm 1) built with neighboring measures has probably no interest.

**Artefacts and numerical risks** By extension, looking for too local fits can become crafty. It is known that data can be slyly laid out to produce excellent numerical fits for the optimization criterion (LMS or LTS), but be physically degenerate<sup>3</sup>. This unfortunate coalition of data can be explained by the nature of Vision models (numerous parameters, non-linearities...).

**Some propositions from literature** Researchers have proposed alternative strategies to the “minimal subsets” in order to avoid a blind sampling process. Veelaert [16] uses rigid sample chains. Braivlovsky proposes a similar idea in a bayesian context [2], but his method cannot be easily generalised. Roth and Levine [4] investigate the recombination of good samples.

The MINPRAN estimator [14] intrinsically leads to fits which are well spread over the (visual) measure space. Bucketing techniques are also used and provide random sampling methods which are physically constrained.

**Robust estimation using buckets** Bucketing techniques use clusters of measures –*buckets*– that cover the whole data space. There are several ways of building subsets of data. Points can be grouped in equally sized subsets. They can also be spread across buckets that split the space arbitrarily along one or several dimensions, as shown in figure 2 (fixed-size buckets are suggested in [18]).



**Fig. 2.** Buckets gathering a defined number of points and dividing space. **Fig. 3.** Two proposed ways of dividing space into 3x4 buckets.

All variations for this method are possible. They allow to divide the visual space without necessarily turning it into a partition (the intersection between two buckets may not be empty).

<sup>3</sup> Stewart [13] talks about “hallucinated fits”.

The sampling stage of the previous algorithm is now directed by buckets. In our work, we build buckets by cutting along dimensions of (input) data. The space is first split along a dimension and then each slice is divided along the other dimensions. Figure 3 shows possible instances of this cutting policy for 2D input data.

## 4 Our working context and our contributions

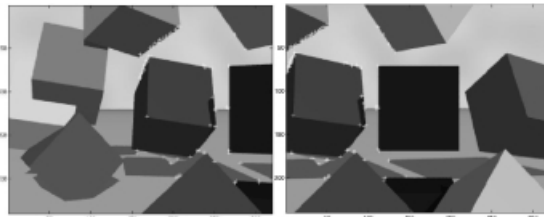
As a revealing example in the Computer Vision field, we investigate an homography estimation problem [9, 7]. In our method, we use a LTS robust estimator for which each sample provides a  $\beta^j$  candidate in the TLS sense. We want to measure the influence of different bucket based clusterings (*cuttings*) on the robust regression process. We also compare three sampling strategies:

- *ssmin*: classical minimal subset strategy (not directed by buckets),
- *ssminBuckets*: a minimum number of points  $n_0$  is sampled, each point belonging to a different bucket [17] (minimal subset strategy among buckets),
- *onePerBucket*: one point is randomly sampled from each bucket (this strategy has not been used before as far as we now).

The onePerBucket strategy seems to be incompatible with robust solving algorithms. Indeed, if one bucket is filled with outliers, all samples will be contaminated. However:

- we propose to use *several* cuttings. Among these cuttings, we hope that one will allow a good *diffusion of outliers among the buckets*. This will permit to find a good sample, for at least one cutting.
- forcing the measures of a sample to cover the bucketed space is a natural way to respect the Computer Vision specific hypothesis (section 3).
- in terms of the bias-variance compromise, we force the exploration to look around a fit which is physically satisfying. We also put limits to the numerical quality of the estimation (giving limits to variance implies increasing the bias).

## 5 Experimental context (homography estimation)



**Fig. 4.** Images and interest points (in white)

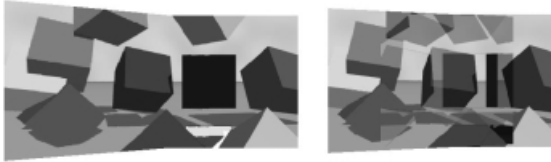
In the homography estimation problem, the  $(\tilde{x}_i, \tilde{y}_i)$  are couples of corresponding points on two images (figure 4). The parameters  $\beta$  of the model are the coefficients of the homography matrix. Following the black box representation:

$$x \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \longrightarrow \beta = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & . \end{pmatrix} \longrightarrow y \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

with

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = f \left( \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right) = \begin{bmatrix} \frac{h_{11}x_1 + h_{12}x_2 + h_{13}}{h_{31}x_1 + h_{32}x_2 + 1} \\ \frac{h_{21}x_1 + h_{22}x_2 + h_{23}}{h_{31}x_1 + h_{32}x_2 + 1} \end{bmatrix}$$

We use synthetic images for which we know the ground-truth parameter vector  $\beta^*$  (corresponding mosaic on left part of figure 5).



**Fig. 5.** *expected homographies : perfect (left) and sly (right)*

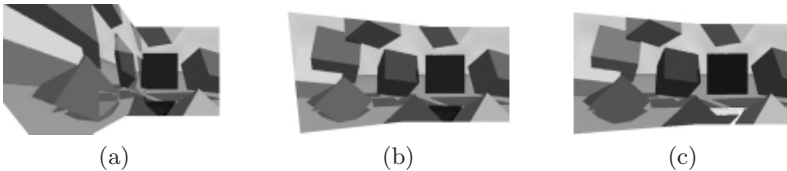
We contaminate the data by introducing 35% of sly points fitting locally to another homography  $\beta^{sly}$  (left part of figure 5). We evaluate the quality of the process with two criteria:

- Best Criterion (*BC*): the optimization criterion (sum of residuals),
- Quality Distance (*QD*): based on the ground-truth,  $QD = \frac{1}{n_r} \sum_{i=1}^{n_r} \|f(x_i^i, \hat{\beta}) - f(x_i^i, \beta^*)\|$ , where  $(x_i^i)_{i=1..n_r}$  are arbitrarily chosen points on the left image.

As presented in section 4, we use different sampling strategies : *ssmin* ( $N = 8000$ ), *ssminBuckets* ( $N = 800$ ) and *onePerBucket* ( $N = 300$ ). For *ssminBuckets* and *onePerBucket*, the cuttings (from the dividing space family) vary from  $2 \times 2$  ( $2 \times 3 \dots$ ) to  $6 \times 6$ .

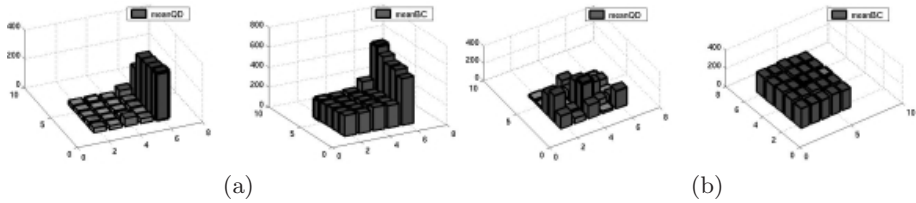
For each strategy,  $N$  has been chosen experimentally to guarantee reproducibility (low variance of *BC* and *QD* around *MeanBC* and *MeanQD* throughout executions, for hundreds of points).

## 5.1 Results



**Fig. 6.** *Representative resulting mosaics for ssmin (a), ssminBuckets (b) and onePerBucket (c) strategies.*

With this corrupted data, `ssmin` gives the worst results. `SsminBuckets` produces homographies strongly affected by the sly data. `OnePerBucket` manages to find good fits (visually coherent homographies, with excellent QD).



**Fig. 7.** Values of the criteria in function of the cuttings (number of  $x$  and  $y$  divisions) in the case of the `onePerBucket` (a) and `ssminBuckets` (b) strategies

**onePerBucket strategy.** Figure 7(a) shows:

- that the `onePerBucket` strategy is sensitive to the different cuttings,
- a correlation between the worst values of MeanBC obtained for  $6 \times x$  and the high values of MeanQD. This can be used to detect which cuttings are the least adapted to the `onePerBucket` strategy.

**ssminBuckets strategy.** We see in figure 7(b) that:

- `ssminBuckets` is less sensitive to the different buckets partitions on MeanBC (low value differences between buckets and low correlation with MeanQD). As for `ssmin`, the optimization criterion LTS is always minimized, even artificially (effect of sly data).
- however, the MeanQD criterion could be revealing (as its values are different among buckets) but it is not available in real problems.

Thus, there is no systematic way for `ssminBuckets` to find a good cutting for data and possibly a good estimation.

## 5.2 Introduction of a diagnostic

Since the `onePerBucket` method is cutting-sensitive, we can easily:

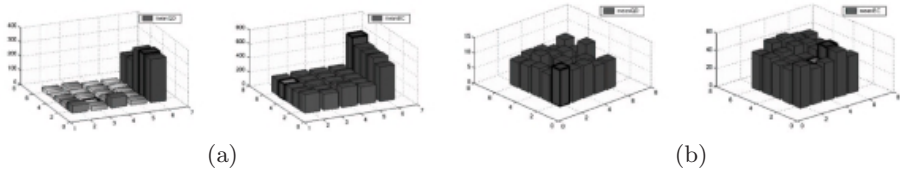
- detect (in the meanBC sense) the correct cuttings by eliminating the worst ones.
- use these best cuttings to eliminate outliers. These outliers are characterized by the biggest cumulated residuals (see the “Median Absolute Deviation” based diagnostic for instance [10]).
- redo the estimation on the epurated data.

We propose a general scheme for an algorithm associated to a diagnostic (figure 8).

1. **Repeat**
  - Build cuttings (from  $2 \times 2$  to  $6 \times 6$  for example).
  - Run onePerBucket strategy (still with LTS based on TLS) with a few iterations.
  - Keep the best cuttings (MeanBC criterion).
  - Eliminate a fixed small fraction of outliers (*rejection* of the points with the biggest cumulated residuals).
- until** solving has become insensitive to the different cuttings
3. Keep the best estimation of this last step.

**Fig. 8.** *Diagnostic algorithm*

**Stopping criterion.** Along the outlier rejection step, onePerBucket strategy becomes less sensitive to the different cuttings (see the evolution of MeanBC among cuttings between 9(a) and 9(b)). At the same time, estimations get better (strong decreasing of MeanQD and MeanBC values between 9(a) and 9(b)). Thus we call the solving *unsensitive* when the variance of MeanBC among buckets drops below a threshold. After one run of algorithm 8, mosaics are perfect.



**Fig. 9.** *onePerBucket : before (a) and after (b) one run of the diagnostic algorithm.*

The efficiency of this algorithm can be verified by introducing gaussian (null mean and variable standard deviation) errors on the coordinates of interest points (figure 4) initially precisely detected and matched. The following table shows the compared efficiency of diagnostics. The measures presented are the percentage of true outliers (known by to the ground-truth) that are correctly identified. These results are averaged over numerous executions.

	$\sigma = 0$	$\sigma = 1$	$\sigma = 2$	$\sigma = 3$
ssmin	64 %	45%	35%	35%
ssminBuckets	60%	51%	53%	38%
onePerBucket	70%	65%	86%	90%

Here again, the onePerBucket strategy is the most efficient. Experiments on real images are also conclusive, and we still work on an automatic parametrization of the diagnostic algorithm.

## 6 Conclusion

We have shown the interest of a resolution strategy based on the use of buckets. We have also proposed a random sampling strategy using “one measure per bucket” in order to physically restrain the exploration. The choice we made to

work on several cuttings has been crucial. This method is fast (less iterations are needed), precise, and robust (sly data is evicted).

We will now focus on decision criterions less directly correlated with optimization criterions in order to identify the best bucket repartition.

In this article, we give conclusions in the case of a LTS estimator and a TLS resolution, but our tests suggest that other estimators behave similarly.

## References

1. P.T. Boggs, R.H. Byrd, and R.B Schnabel. A stable and efficient algorithm for non linear orthogonal distance regression. *SIAM Journal on Scientific and Statistical Computing*, pages 1052–1078, 1987.
2. V. Bravilovsky. An approach to outlier detection based on a probalistic model. *ICPR'96, Vienna*, 2:70–75, 1996.
3. M.A. Fischler and R.C. Bolles. Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography. *Communications for the Association for Computing Machinery*, pages 381–895, 1981.
4. M.D. Levine G. Roth. Geometric primitives extraction using a genetic algorithm. *IEEE transactions on Pattern Analysis and Machine Intelligence*, pages 901–905, 1994.
5. S. Van Huffel. *Recent advances in total least squares techniques and error-in-variables modeling*. SIAM, 1997.
6. S. Van Huffel and J. Vandewalle. *The total least squares problem, computational aspects and analysis*. SIAM, 1991.
7. B. Thiesse M. Douze, V. Charvillat. More precise and robust mosaics. In *proceedings of RFIA*, volume 1, Angers, France, 2002.
8. P. Meer, D. Mintz, A. Rosenfeld, and D.Y. Kim. Robust regression in computer vision : a review. *International Journal in Computer Vision*, pages 59–71, 1991.
9. A. Zisserman R. Hartley. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
10. P.J. Rousseeuw and K. Van Driessen. Computing lts for large data sets. *Tech. Report. Univ. of Antwerp*, 1999.
11. P.J. Rousseeuw and M. Hubert. Recent developments in progress. *L1 - Statistical procedures and related topics*, 1997.
12. P.J. Rousseeuw and A.M. Leroy. *Robust regression and outlier detection*. John Wiley and Sons, 1987.
13. C.V. Stewart. A new robust operator for computer vision : theoretical analysis. Technical report, Departement of Computer Science, Rensselaer, NY, 1993.
14. C.V. Stewart. Minpran : a new robust operator for computer vision. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 1995.
15. P.H.S Torr and D.W. Murray. The developpement and comparison of robust methods for estimating the fundamental matrix. *Int. Journal of Computer Vision*, 24(2):271, 1997.
16. P. Veelaert. Constructive fitting and extraction of geometric primitives. *Graphical Models and Image Processing*, pages 233–251, 1997.
17. Z. Zhang. Tutorial on parameter estimation techniques. <http://www-sop.inria.fr/robotvis/personnel/zzhang/Publis/Tutorial-Estim/node25.html>, 1996.
18. Z. Zhang. Parameter estimation techniques : a tutorial with application to conic fitting. *Image and Vision computing*, pages 59–76, 1997.