MVA2009 IAPR Conference on Machine Vision Applications, May 20-22, 2009, Yokohama, JAPAN

Linear 3-D Object Pose Estimation with Dense Sample Images

–Discussions about Limitation of Parameter Estimation Ability by the Linear Regressions–

Toshiyuki Amano
Graduate School of Information Science,
Nara Institute of Science and Technology
amano@is.naist.jp

Toru Tamaki
Graduate School of Engineering
Hiroshima University
tamaki@hiroshima-u.ac.jp

Abstract

In the image parameter estimation by the linear regression, it has very high degrees of freedom for the decision of regression coefficients, because the dimension of image vector is huge high. In this paper, we discuss its potential by the learning of the dense samples. For the learning process, we employed a sequential regression coefficient calculation algorithm and realize its calculation for dense samples with reasonable computational cost. Through the experimental result, we discuss about the limitation of parameter estimation ability by the linear regression.

1 Introduction

The appearance based pose estimation method represented by the parametric eigen space method [1, 2] does not need any feature extraction for parameter estimation, and the method is widely applied by many applications for its convenience [3, 4, 5]. The parametric eigen space method estimates object's pose by the nearest neighbor search between parametric manifold and projection point of an input sample on the eigen space. However, we can implement by the simple linear regression instead of such parameter estimation [7].

The regression based method is not limited in the linear regression, non linear regression based on kernel-CCA [8], kernel-SVM [9] are proposed. These kernel based methods are powerful and flexible since they find an optimal curved subspace. However, the image vector has vast degrees of freedom for decision of the regression coefficients even if we use linear regression, since the explanatory variables are corresponding to image pixels(Figure 1). The N dimensional image vector space is capable the loss less regression for $N-1$ learning samples. We can not decide the optimal regression coefficients without its criterion from the sparse samples, but if we have dense learning samples, the optimal regression coefficients must be fixed. The dense sample learning gives continuous movement of the image vector for each parameters and if we can regard its movement as linear relation at the all local regions by the parameters, its coefficients give a perfect regression for the parameter estimation by the linear regression. It means the degree-of-freedom of the manifold in the image vector space not exceeds the dimension of the image vector.

In this paper, we employ a sequential update scheme of regression coefficients for the dense learning samples and discuss about the limit of the parameter estimation ability by the linear regressions with the experimental results.

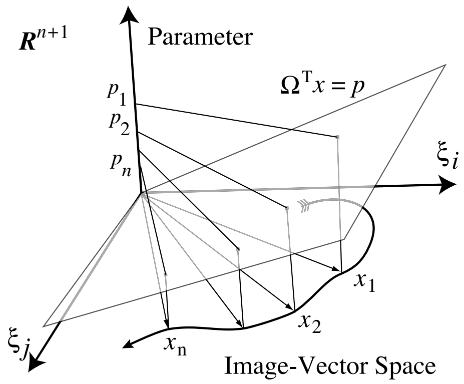


Figure 1: Capability of the N dimensional vector space

2 Sequential Update Scheme

Nowadays, most of PC has several hundred GB external memory (HDD) and a few GB main storage (RAM), even if it is low price PC. Thus, to store dense sample by the external memory is easy, but to calculate coefficients of linear regressions for parameter estimation is still hard. When dealing with linear regression problems, we can calculate regression coefficients Ω for sample images $\{x_i | i = 1, 2, \ldots, n\}, x_i \in R^N$ and image parameter $\{p_i | i = 1, 2, \ldots, n\}, p_i \in R$ such as

$$p_i = \Omega^T x_i \tag{1}$$

by the Moore-Penrose matrix inverse

$$\Omega = X(X^T X)^{-1} P, \tag{2}$$

where $X = [x_1, x_2, \ldots, x_n]$, $P = [p_1, p_2, \ldots, p_n]$, n is the number of samples and N is equivalent to the number of pixels. In most case of regression problem of the image parameter estimation, $n \ll N$ and the calculation of eq.(2) is easy if n is less than a few hundred. However, if we had a full number of samples that means $n = N$ for the dense samples of regression, we need over a few GB main memory for the calculation of coefficient for the 128×128 gray scale image. More over, the calculation cost of matrix inverse is not reasonable. Therefore, we employ the regression coefficient calculation method by the sequential update.

The first, we determine an initial coefficient Ω_1 that satisfies eq.(1) for a sample x_1 that has parameter p_1 by

$$\Omega_1 = k_1 u_1 \tag{3}$$

$$u_1 = \frac{1}{|x_1|} x_1 \tag{4}$$

where $k_1 = p_1/|x_1|$. Next, we calculate the orthogonal component of x_2

$$u_2 = \frac{1}{|u_2'|} u_2' \tag{5}$$

$$u_2' = x_2 - (u_1^T x_2) u_1 \tag{6}$$

that is orthogonal to u_1 and we determine the coefficient Ω_2 for the samples x_1 and x_2 by the k_2 that satisfies

$$p_2 = (\Omega_1 + k_2 u_2)^T x_2 = \Omega_2^T x_2 \tag{7}$$

According to this scheme, ith normal vector u_i is calculated by Gram-Schmidt orthogonalization of x_i with $\{x_j | j = 1, 2, \ldots, i-1\}$ such as

$$u_i = \frac{1}{|u_i'|} u_i' \tag{8}$$

$$u_i' = x_i - \sum_{j=1}^{i-1} (u_j^T x_i) u_j \tag{9}$$

and we get the ith regression coefficient by

$$\Omega_i = \Omega_{i-1} + \frac{1}{u_i^T x_i} (p_i - \Omega_{i-1}^T x_i) u_i. \tag{10}$$

In the implementation of this calculation scheme, we do not have to keep all samples to up date Ω_i. Thus, if we use sequentially reading procedure of u_1 from external memory, the calculation for the full sample of coefficient is possible by reasonable cost. For example, we can calculate the full sample regression coefficient of double precision arithmetic for 128×128 pixels gray scale images by a few hundred KB main storage and 2.1GB external memory.

3 Experiment

3.1 Object Recognition with Pose Estimation by the Linear Regression

In this section, we apply the sequential update scheme to the object recognition and 1-DOF pose estimation problem. For test samples, we use COIL-20 [6] that is including 72 images for 20 objects at varying pose to corresponding pose parameters of yaw angle. Thus, the total number of this library is 1440 and each image is 128×128 pixels gray scale. In this experiment, we put the regression coefficients Ω_{obj},Ω_c and Ω_s that satisfy

$$obj = {\Omega_{obj}}^T x \tag{11}$$

$$\cos(\theta) = {\Omega_c}^T x \tag{12}$$

$$\sin(\theta) = {\Omega_s}^T x \tag{13}$$

for the object number obj and its pose angle θ estimation. We use extended posture expression on unit circle, because θ is the cyclical parameter. For the object recognition, we assign $obj = \{1, 2, \ldots, 20\}$ for each object and recognize the object number by the rounding of estimated obj.

The regression coefficients at each update step are shown in figure 2. For the update process, we chose the

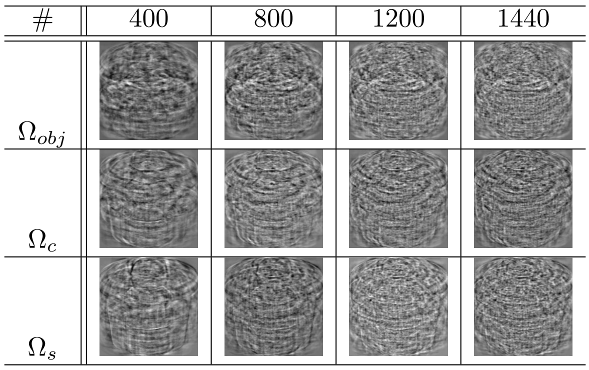
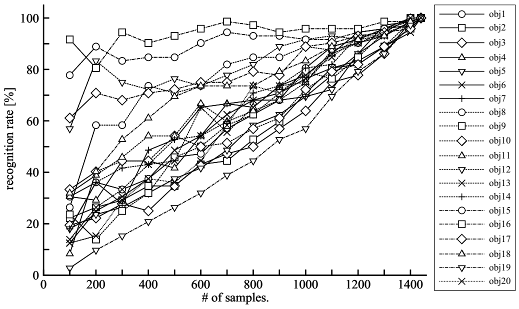
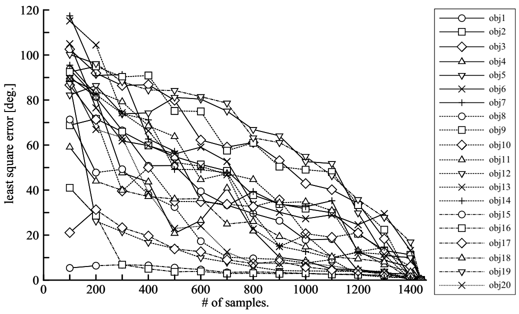


Figure 2: Regression Coefficients



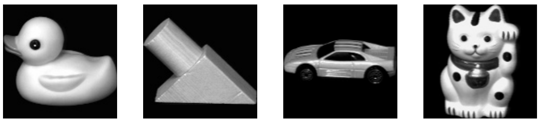
(a) Object Recognition Results.



(b) Pose Estimation Results.

Figure 3: Parameter Estimation Results.

ith learning sample x_i by the random sampling from 1440 samples without duplicated sample. The parameter estimation results by these coefficients are shown in figure 3. The vertical axes show the recognition rate and pose-estimation error by the RMSE at each object. The horizontal axes are the number of learning samples for the coefficient calculation that is meaning the coefficients of each update steps. The evaluation is done by all samples. Thus, the result at 1440 is meaning the evaluation by the closed samples. From these results, we can see the recognition error and pose estimation error are reducing along with increasing of the number of learning samples. In the evaluation with the average by all objects, the recognition rate and pose estimation error were 68.6% and 30.4[deg.] by the coefficients calculated by 800 samples, and were 100% and 1.18×10^{-4}[deg.] (it is rounding error) by all samples. Therefore, this simple linear regression has capacity to estimate the parameters without error for 1440 samples of parameter estimation problem since $dim(x_i) = 128^2$ is very huge compared to 1440. The computational



object 1 object 2 object 3 object 4

Figure 4: Example images of 4 Objects.

time for regression coefficients was 161 sec. by the linux based PC (CPU:2.67GHz, RAM:2GB).

3.2 2-DOF Pose Estimation

In the object recognition and 1-DOF pose estimation, its estimation was easy even if we used linear regression since the samples are not dense compared to the dimension of image vector. In this section, we apply linear regression to the dense samples that have a 2-DOF parameter of posture angles. For the 2-DOF rotation sample, we put the roll angle ψ for the 2nd parameter in addition to yaw angle θ and generate the variation for ψ by the in-plane rotation of image plane. We set the step parameter as $\{(\theta,\psi)|\theta = 0, 5, \ldots, 355[deg.], \psi = 0, \Delta\psi, \ldots, 255\Delta\psi, \Delta\psi = 360/255[deg.]\}$ and we calculate the linear regression coefficients for the four objects shown in figure 4 such as

$$\cos(\theta) = {\Omega_c^\theta}^T x \tag{14}$$

$$\sin(\theta) = {\Omega_s^\theta}^T x \tag{15}$$

$$\cos(\psi) = {\Omega_c^\psi}^T x \tag{16}$$

$$\sin(\psi) = {\Omega_s^\psi}^T x \tag{17}$$

with the sequential update scheme. Thus, the total number of learning samples is 18360 for each object and that is greater than $dim(x_i) = 16384$. The figure 5 shows the coefficients for the object 4 at the update step of 1000, 4000, 10000, 14800. In this experiment, the update was done by x_i that sampled from 18360 learning samples by random sampling without duplication at each object. The 2-DOF pose estimation results for each object are shown in the figure 6. The vertical axis is a pose estimation error of RMSE that evaluated by 18360 samples for each object, and the horizontal axis is the number of samples for coefficients calculations. From the figure 6, we can see the error is reducing along with increasing of samples for coefficient calculation as the general trend of pose estimation error. However, the estimation error is rising and rapidly increasing at the last part. In case of object 4, estimation error is increasing around over 12000 samples gently and diverging at the 14800 samples. We think this reason is related to the failure of the calculation of regression coefficients. It is confirmed since these coefficients at the 14800 steps are flattened. The further detail of this mechanism is discussed at the next section. The computational time of the regression coefficients for the object 4 was 4 hour 51 min. by the linux based PC (CPU:2.67GHz, RAM:2GB).

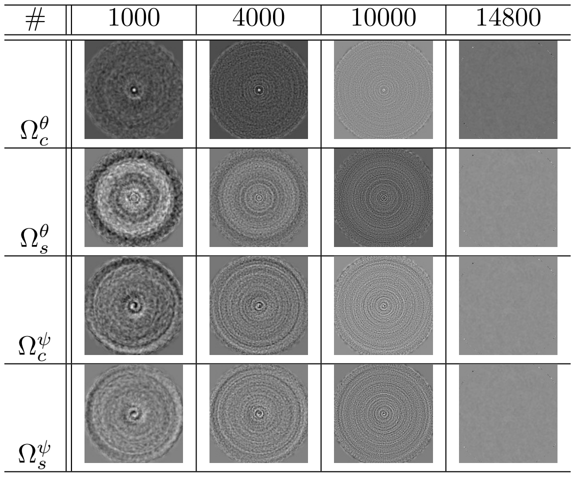


Figure 5: Regression Coefficients (obj4).

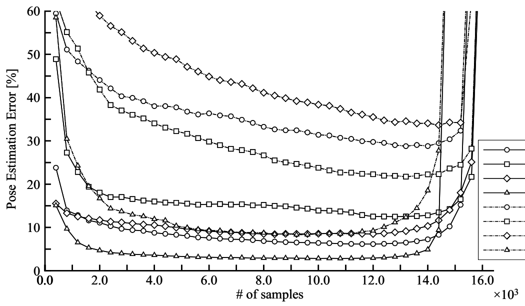


Figure 6: 2-DOF Pose Estimation Error.

4 Discussion

4.1 Condition of Error-free Estimation

Eventually, any linear parameter estimation method requires linear relation between the distribution of the samples on the parameter space and image vector space. In other words, any unsupervised sample (y, q) has to comply to

$$q = \Omega^T y \tag{18}$$

where Ω is regression coefficients calculated by learning samples $\{x_i|i = 1, 2, \ldots, n\} \in \mathcal{R}^N$. It means the vector $(y^T, q)^T$ that is supervised sample has to lieing on the hyper plane that is spanned by learning samples $\{(x_i^T, p_i)^T|i = 1, 2, \ldots, n\}$ in the $\mathcal{R}^{N+1}$. If the sample has this property, we can see the convergence of the coefficient Ω_i at the update step $k \leq n+1$ and the samples satisfy $Rank(\{(x_i^T, p_i)^T|i = 1, 2, \ldots\}) = k$. However, the calculation of linear regression fails in case of $Rank(\{x_i|i = 1, 2, \ldots\}) < k$. Thus, we get the condition of the image parameter estimation without estimation error of the linear estimation as

$$Rank(\{(x_i, p_i)\}) = Rank(\{x_i\}) \leq n \leq N'. \tag{19}$$

Where N' is the number of valid pixels that expressed following section. From the figure 3, we can see the estimation error is remaining at the pose estimation with the coefficients calculated by the samples less than 1440. Therefor, the coefficient is still not converging, but the dimension of x_i is more than 10 times higher

Table 1: Valid Pixels.

object1	object2	object3	object4
15,928	16,058	16,271	14,780

compared to 1440 and we can apply more over 10 times dense learning. In the future work, we wish to explore whether the convergence of Ω_i exist or not, and we wish to make its condition clear.

4.2 The Number of Upper Limit

If the samples $x_i \in \mathcal{R}^N$ are independent each other, the parameter estimation has capability to make mapping up to N pieces of the samples. However, we can see the parameter estimation failure at the 14800 to 15600 samples from the figure 6. We think the cause of this estimation failure is related with the number of valid pixels. In the figure 5, we can see the flat parts at the four corners that are outside of circular part at the coefficients. These parts are corresponding to the background, and the element values in its part are constant for any parameters. Therefore, these pixels are not contributed as explanatory variable. We show the number of valid pixels that contributed as explanatory variable for each object in the table 1. These values are corresponding with the number of samples, and the pose estimations fail in these values. Therefore, we think the upper limit of the number of the samples is decided by the valid pixels. However, the gradual increase of the estimation error before valid pixel is still mystery. The curse of dimension is well known, but the cause of this phenomenon is not depending since employed sequential update scheme has no approximation by the subspace. In the future work, we wish to make this mystery.

4.3 The Accuracy Decrement with The Increase of the Number of Samples

In the result that is shown in section 3.1, the accuracy of the parameter estimation is raised with increasing of the number of samples. However, the accuracy was dropped with increasing of the number of samples in case of section 3.2. Its difference might be explained with the remaining degrees of freedom of invariant projection. In the two DOF parameters estimation problem of θ and ψ, the θ estimation is written as

$$\cos(\theta) = {\Omega_c^{\theta}}'^{T} G_\psi x \tag{20}$$

$$\sin(\theta) = {\Omega_s^{\theta}}'^{T} G_\psi x \tag{21}$$

by using invariant projection G_ψ for ψ. This means the input image is converted to the in plane rotation invariant image for the continuous rotation by ψ and independent component is only r pixels that is the length of radius. Therefore, the parameter estimation is failed by more than r variations of θ. In its experiment, the number of step of psi was 256 and the image rotation is able to regard as smooth change. Therefore, its remaining degree of freedom was only 64 and it is less than 72 of theta variations. We think the reason of the accuracy drop is connected to this, but it is not confirmed yet. In future work, we are going to make this relation clear.

5 Conclusion

In this paper, we employed the sequential update scheme for the calculation of regression coefficients. We apply this scheme to image parameter estimation with the dense samples and showed the parameter estimation ability by the linear regression. Through the experimental results, we showed the condition of the error-free estimation by the linear regression. In addition, we confirmed the linear parameter estimation method has the capacity to learn the number of samples up to the number of valid pixels. However, existence of coefficient convergence and the cause of gradual increase of the estimation error are still mystery. We wish to figure out these mysteries in the future work.

References

[1] H. Murase and S. K. Nayar, “Visual learning and recognition of 3-D objects from appearance,” *Int. J. Comput. Vision*,vol.14, no.1, pp.5–24, 1995.

[2] S. K. Nayar, H. Murase, S. A. Nene, “Parametric Appearance Representation,” *Early Visual Learning*, Chapter 6, pp.131?160, Oxford University Press, 1996.

[3] H. Murase, S. K. Nayar, “Illumination Planning for Object Recognition Using Parametric Eigenspaces,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.16, No.12, pp.1219?1227, 1994.

[4] H. Murase, R. Sakai, “Moving object recognition in eigenspace representation: Gait analysis and lip reading,” *Pattern recognition letters*, Vol. 17, pp.155?162, 1996.

[5] K. Deguchi, and T. Noguchi, “Visual servoing using eigenspace method and dynamic calculation of interaction matrices,” *Proc. of the 13th International Conference on Pattern Recgnition*, Vol.1,pp. 302?306,1996.

[6] S. Nene and S. Nayar and H. Murase, “Columbia Object Image Library: COIL-20,” *Technical Report CUCS-006-96*, Department of Computer Science, Columbia University, 1996.

[7] T. Okatani and K. Deguchi, “Yet Another Appearance Based Method for Pose Estimation Based on a Linear Model,” *Proc. of IAPR Workshop on machine vision applications*, pp.258–261,2000.

[8] T. Melzer and M. Reiter and H. Bischof, “Appearance models based on kernel canonical correlation analysis,” *Pattern Recognition*,vol.36,pp.1961–1971,2003.

[9] S. Ando and Y. Kusachi and A. Suzuki and K. Arakawa, “Appearance based pose estimation of 3D object using support vector regression,” *International Conference on Image Processing*,vol.1,pp.34–4,2005.