A Combined Probabilistic Framework for Learning Gestures and Actions

Francisco Escolano, Miguel Cazorla, Domingo Gallardo, Faraón Llorens,
Rosana Satorre and Ramón Rizo

Grupo i3a: Informática Industrial e Inteligencia Artificial
Departamento de Ciencia de la Computación e Inteligencia Artificial
Universidad de Alicante
E-03690, San Vicente, Spain
Fax/Phone: 346-5903681
e-mail: sco@i3a.dtic.ua.es

Abstract. In this paper we introduce a probabilistic approach to support visual supervision and gesture recognition. Task knowledge is both of geometric and visual nature and it is encoded in parametric eigenspaces. Learning processes for compute modal subspaces (*eigenspaces*) are the core of tracking and recognition of gestures and tasks. We describe the overall architecture of the system and detail learning processes and gesture design. Finally we show experimental results of tracking and recognition in *block-world like* assembling tasks and in general human gestures.

Keywords. Visual Inspection, Gesture Recognition, Learning, Probabilistic Constraints, Eigenmethods

1 Introduction

Eigenspace representations obtained with Principal-Component Analysis [12] provide compact representations of both visual appearance (color and texture) and object geometry (rigid and non-rigid deformations). These models capture the main characteristic variability in spatial and temporal domains. They are useful for general gesture recognition [3]. View-based approaches rely on image models. Spatial variability is used to model human face appearance [18] and [15]. The relation between pose and appearance parameters is studied in [17]. Robust models are described in [2]. Lips motion models [13] and general gestures [16] [8] [7] are based on temporal variability. On the other hand geometric eigenmodels are applied to model object deformations: in [19] natural shape recognition is based on eigenmodels, and, finally in [4] [5] [14] deformable model fitting is driven by projecting shape information in low-dimensional spaces. Appearance and geometric information is integrated in [6]. We propose a gesture tracking and recognition system which is based on geometric and visual appearance. The key question is to combine several sources of variability (eigenspaces). These modal spaces are the core of the system. In the first section we define a general gesture model. In the second we present the modules of the system. Learning processes and eigenspaces are defined in the third section. Finally we present tracking and recognition results.

2 Gesture Models and Perceptual Processes

2.1 Gesture Models: Objects and Constraints

An *Action/Gesture* model $\mathcal{M} = [\mathcal{O}(t), C(t)]$ defined $\forall t \in T$ over a *temporal window* $T = [t_s, t_e]$ consists of two basic elements :

1. *Object/Entity Set:* $\mathcal{O}(t) = [\mathcal{T}_i(t), \mathcal{P}_i(\Phi(t))]_{i=1}^{g}$ which parametrically defines the objects, entities or regions of interest for the task of visual supervision. We consider two types of objects: *primary* (reference objects) and *secondary*. For each object we must specify time-dependent parametric functions which characterize:
   (a) *Geometric Appearance:* $\mathcal{T}(t) = [\Theta_M(t), \Theta_P(t)]$ includes morphological[1] parameters $\Theta_M(t)$ and positional/affine [2] parameters $\Theta_M(t)$.
   (b) *Visual Appearance:* $\mathcal{P}(\Phi(t)) = [\Theta_I(t)]$ associates a *characteristic brigthness pattern* [3], defined by the parameters $\Theta_I(t)$, to each object or entity; and incorporates a *time-warping function* $\Phi(t)$ for compensating time-delay effects.
2. *EigenConstraint Set:* $C(t)$ defines *spatio-temporal bounds* over geometric and appearance parameters. These bounds are of stochastic nature and can be:
   (a) *Absolute Constraints:* $C_{abs}(t) = [\mathcal{A}_j(t)]_{j=1}^{a}$ which are associated to primary objects and locally constrain the morphlogical or positional evolution. They are denoted are by $\mathcal{A}(\Theta_M(t))$ in the first case and by $\mathcal{A}(\Theta_P(t))$ in the second.
   (b) *Relative Constraints:* $C_{rel}(t) = [\mathcal{R}_k(t)]_{k=1}^{r}$ are denoted by $\mathcal{R}(\varphi(\Theta_P(t)))$ and relates with $\varphi(\Theta_P(t))$ positional parameters of primary objects with parameters, of the same type corresponding to secondary objets.
   (c) *Appearance Constraints:* $C_{app}(t) = [\mathcal{P}_{Wl}(t)]_{l=1}^{a}$ associated to every object and denoted by $\mathcal{P}_W(\Theta_I(t))$ and define intensity models for performing texture/color segmentation and classification.

Each gesture or action is then characterized by the spatio-temporal variation of both affine and free-form parameters associated to the objects of the structure.

[1] Local deformations.
[2] Translation, rotation and scale.
[3] Light effects, color, texture.

2.2 Perceptual Processes: from Learning to Recognition

Assuming this geometric approach, the task of extracting usefull medium-level information from image sequences can be performed by a set of *Perceptual Processes* capable of learning, tracking and recognizing geometric spatio-temporal paths. These processes, which rely on a *modal based* approach, can be categorized as follows:

1. *Learning Processes:* for deriving the characteristic constraint parameters by means of extracting and analyzing their associated eigenspaces with quality $\zeta \in [0,1]$ which indicates the proportion of variability considered:
   (a) `:learn-geometric`($\mathcal{S}_G(t), \mathcal{A}(t)/\mathcal{R}(t), \zeta$) extracts an absolute $\mathcal{A}(t)$ or relative $\mathcal{R}(t)$ modal space from a geometric training set $\mathcal{S}_G(t)$.
   (b) `:learn-color`($\mathcal{S}_C(t), \mathcal{P}_W(t), W, \zeta$) once a size mask W is defined, and given a color training set $\mathcal{S}_C(t)$ this process computes a RGB parametric space $\mathcal{P}_W(t)$ which contains the color specifications of a region.
2. *Segmentation and Tracking Processes:* color segmentation, prediction and estimation of local and structural configurations is guided by the knowledge previously learnt:
   (a) `:filter-color`($I(\Phi(t)), \mathcal{P}_W(t)$) peforms local color segmentation over $I(\Phi(t))$ by mask convolution (eigenspace projection and color recognition). The result is a binary image $\mathcal{P}(\Phi(t))$ which is processed with morphological filters (opening and closing).
   (b) `:track-local`($\mathcal{T}'(t), \mathcal{P}(\Phi(t))$) once color segmentation is performed, this process locally computes morphological and positional parameters of a primary object without applying constraints.
   (c) `:predict-local`($\mathcal{T}'(t), C_{abs}(t)$) uses absolute constraints to predict the future configuration of a primary objects. If this prediction fails we must apply free local tracking.
   (d) `:track-global`($\mathcal{O}(t), C_{abs}(t) \cup C_{ret}(t)$) combines the absolute prediction with the relative constraints to perform structural or coupled tracking [11].
3. *Recognition/Interpretation Processes:* once tracking is performed the parametric spatio-temporal path is estimated. Then we use the Mahalanobis distance metric, with quality ζ, for recognition:
   (a) `:satisfy-constraint`($\mathcal{O}(t), \mathcal{A}(t), \zeta$) a constraint is satisfied by an structure or object set if the projection of the corresponding parameters over the eigenspace falls inside the admissible probabilistic limits.
   (b) `:satisfy-all`($\mathcal{O}(t), C_{abs}(t) \cup C_{ret}(t)$) an object set fits a gesture model if the number of satisfied constraints is greater than a threshold.

3 Learning Processes and Gesture Models

3.1 EigenConstraints

Learning processes which are performed *off-line* are the core of the system. Their purpose is to compute *EigenConstraints*, i.e. modal spaces which capture the main variability from the covariance matrix of each training set. Constraint definition, and hence gesture design, includes serveral specifications. The processes `:learn-geometric` and `:learn-color` must take into account the following items:

: `Constraint` type (absolute, relative or appearance).
: `Parameter Type` which can geometric (morphological, positional) or visual (representing color or texture).
: `Parameters` for representing contours (morphological), center coordinates, angles, scales (absolute positiona), distances, relative angles, relative scales (relative position) and intensity patches (color or texture).
: `Dimension` or number of parameters defined in the original parametric space. The computation of eigenspaces usually induces a reduction of this dimension.
: `Scope` which can be static (time-independent variability) or dynamic (time-dependent variability).
: `Envelope` which is the degree of genericity (high variability/acceptance limits induce general constraints and low limits define more specific constraints).
: `Quality` which is number of modes of variation considered (few modes if variability is concentrated and more modes otherwise).
: `Role` which can be local (the scope is an individual object) or structural (associated to a group of objects).

In Table 1 we have presented all possible types of constraint which can be defined. The parameters that support these constraints are listed in Table 2.

3.2 Gesture Design: Combined Variability

The effect of these considerations, specially the scope and role, depends on the task to be supervised, the types of objects involved and their relationships. The key point is to combine different sources of variability for efficient and complete gesture design. General principles of gesture design are listed below:

1. *Morphological* constraints are individually assigned to each object (they are local). It is assumed a static scope if spatio-temporal shape variability is too low (near constant) or too high (it is not possible to obtain a well defined temporal path).

Table 1. Gesture specification. Types of EigenConstraints

Constraint	Parameter Type	Parameters	Dimension	Scope	Role
$\mathcal{A}$	Morphological	$[\Theta_M(t)]_{t=t_0}^{tf}$	$2 \times p \times T/\Delta t$	Dynamic	Local
	Morphological	$[\Theta_M]_{t=t_0}^{tf}$	$2 \times p$	Static	Local
	Positional	$[\Theta_P(t)]_{t=t_0}^{tf}$	$\{2T, T\}/\Delta t$	Dynamic	Local
	Positional	$[\Theta_{Pi}(t)]_{i=1,t=t_0}^{g,t_f}$	$g \times \{2T, T\}/\Delta t$	Dynamic	Global
	Positional	$[\Theta_{Pi}]_{i=1,t=t_0}^{g,t_f}$	$\{2 \times g, g\}$	Static	Global
$\mathcal{R}$	Positional	$[\varphi(\Theta_P(t))]_{t=t_0}^{t_f}$	$T/\Delta t$	Dynamic	Local
	Positional	$[\varphi_k(\Theta_P(t))]_{k=1,t=t_0}^{\binom{g}{2},t_f}$	$\binom{g}{2} \times T/\Delta t$	Dynamic	Global
	Positional	$[\varphi_k(\Theta_P)]_{k=1}^{\binom{g}{2}}$	$\binom{g}{2}$	Static	Global
$\mathcal{P}$	Texture	$[\Theta_I(t)]_{t=t_0}^{t_f}$	$3 \times W \times T/\Delta t$	Dynamic	Local
	Color	Θ_I	$3 \times W$	Static	Local

Table 2. Parameters/Support for EigenConstraints

Constraint	Parameter Type	Parameters
$\mathcal{A}$	Morphological	$\Theta_M = [x_i, y_i]_{i=1}^{p}$
	Translation	$\Theta_{P_t} = [t_{xi}, t_{yi}]_{i=1}^{g}$
	Scale	$\Theta_{P_s} = [s_{xi}, s_{yi}]_{i=1}^{g}$
	Rotation	$\Theta_{P_\theta} = [\theta_i]_{i=1}^{g}$
$\mathcal{R}$	Distance	$\varphi_k(\Theta_{P_t}) = [\|\|(t_{xi}, t_{yi}) - (t_{xj}, t_{yj})\|\|^2]_{k=1}^{\binom{g}{2}}$
	Scale	$\varphi_k(\Theta_{P_s}) = [\|\|(s_{xi}, s_{yi}) - (s_{xj}, s_{yj})\|\|^2]_{k=1}^{\binom{g}{2}}$
	Angle	$\varphi_k(\Theta_{P_\theta}) = [\|\|(\theta_i - \theta_j)\|\|^2]_{k=1}^{\binom{g}{2}}$
$\mathcal{P}$	Color	$\Theta_I = [r_l, g_l, b_l]_{l=1}^{W}$

2. *Absolute Positional* constraints can be local if they are assigned to reference objects and global if they are associated to a group of objects. In this case it is not necessary to formulate relative constraints, although it can be used to enforce absolute constraints, but spatial invariance is not considered. However, grouping with absolute constraints compensates delays due to individual objects and simplifies tracking.

3. *Relative Positional constraints* are usually global and, in this case, they include parameters extracted from pairs of objects. It is interesting to apply these constraints in combination with absolute constraints, associated to reference objects, in order to simplify tracking processes (coupled tracking). These constraints introduce spatio-temporal invariance.

4. *Appearance constraints* are always local. They are dynamic when we are interested in using texture variation for recognition and static otherwise (use color to identify regions of interest).

4 Tracking and Recognition Examples

4.1 Tracking of Visual Tasks

We have defined spatio-temporal constraints for tracking a *block-assembling task* which consists of pushing four coloured blocks (objects) following a specific order and assumning uniform speed. Considering $T = 35$ frames and $\Delta t = 1$ and using a *robust super-elipsoidal local tracker* [9] the assembling gesture is described in Table 3. Scale and and color parameters are considered near constant along the sequence. The size of color space is 5×5 (75 RGB parameters) and the morphological filter is an opening with a squared estructuring element of size 3×3. Form is modelled by a shape parameter so it is no necessary to compute non-rigid eigenspaces. Position changes are modeled with trajectories and relative distances used to enforce coupled tracking. Tracking results are showed in Fig. 1,Fig. 2 and Fig. 3.

Table 3. Gesture specification. Block Assembling Task

Constraint	Parameter Type	Quality	Scope	Role
Absolute	Positional (translation)	0.9	Dynamic	Global
Absolute	Positional (rotation)	0.9	Dynamic	Global
Absolute	Positional (scale)	0.9	Static ($\approx$ 0.0)	Global
Relative	Distance	0.9	Dynamic	Global
Appearance	Color	1.0, 0.95	Static ($\approx$ 0.0)	Local

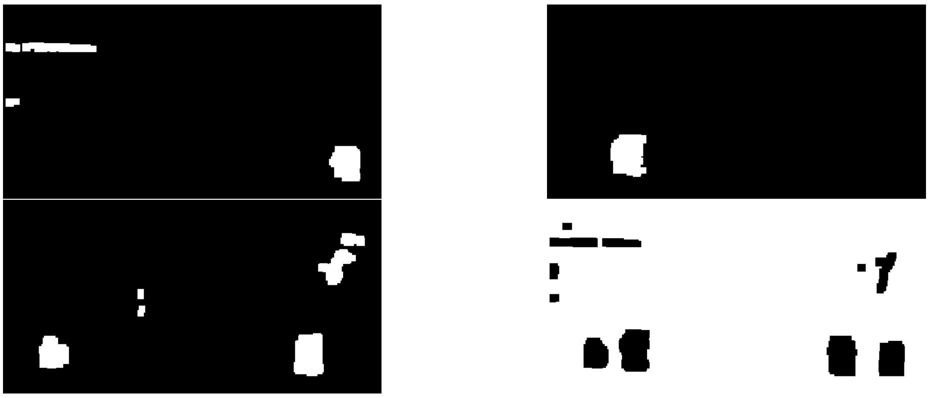

Fig. 1. RGB Color Segmentation.

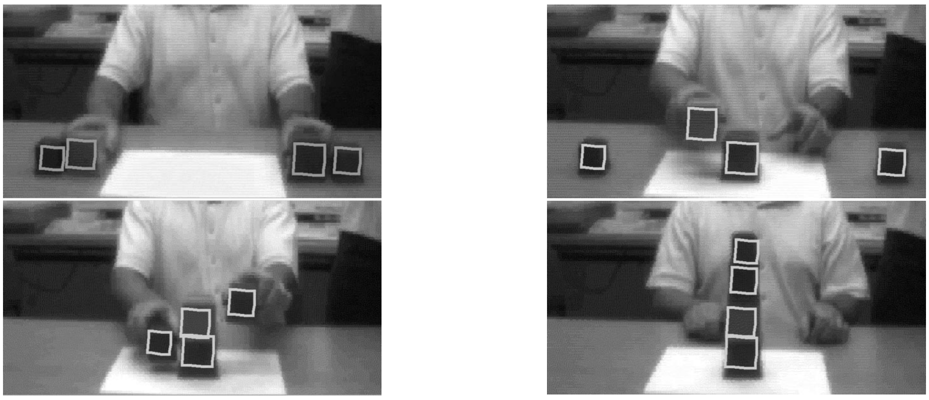

Fig. 2. Coupled Tracking. Result with a good input sequence: the camera follows both the right pushing order and moving speed.

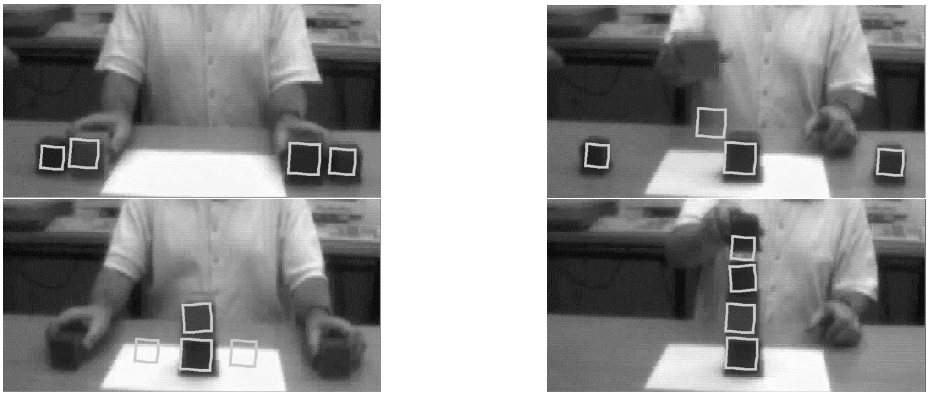

Fig. 3. Coupled Tracking. Result with a bad input sequence: blocks are pushed in the right order but there is a high delay (lookahead) in the first phase of the sequence.

4.2 Tracking and Recognizing Human Gestures

Finally we present another example from human gesture tracking. In this case objects are templates are associated to the head and hands. In this case $T = 10$ frames and $\Delta t = 1$ and a *robust elliptical template* [10] model suffices. In this case only absolute constraints are used. Scale is considered near constant along the sequence. Position changes are modeled with trajectories. Grey segmentation and morphological filters are applied so we can avoid computing color eigenspaces. Tracking results are showed in Fig. 4 and Fig. 5. We have learnt two gesture models. Both gestures are described in Table 4. In the first one the right hand follows a parabolic motion. In the second model this motion is linear. When the input sequence, which satisfies the first model, is presented, it will be recognized by the first tracker because the number of satisfied constraints will be greater than the number of satisfied constraints in the second case. If the envelope of these constraints is too high this input will be recognized by both models.

Table 4. EigenConstraints specification. Human Gesture

Constraint	Parameter Type	Quality	Scope	Role
Absolute	Translation	0.9	Dynamic	Global
Absolute	Rotation	0.9	Dynamic	Global
Absolute	Scale	0.9	Static (≈ 0.0)	Global

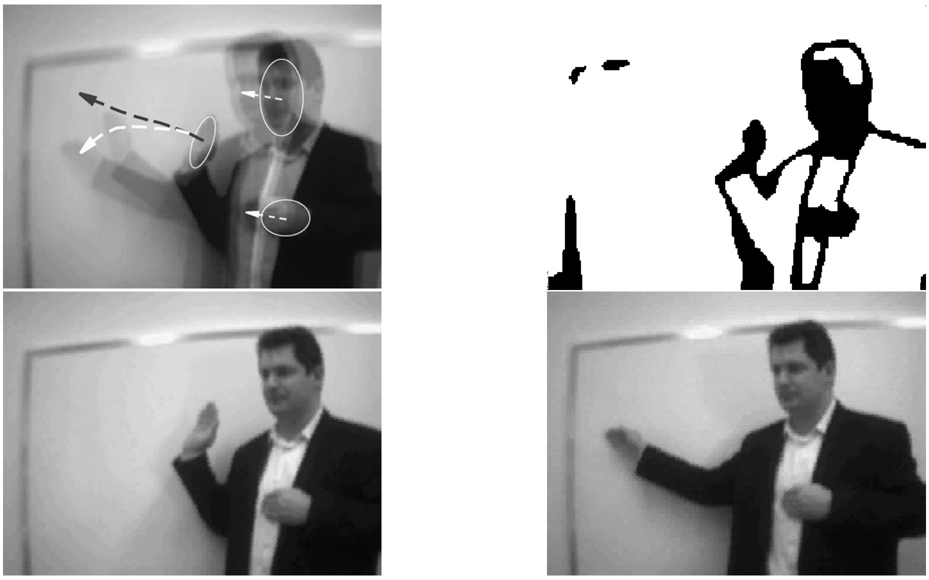

Fig. 4. Human Gesture Tracking. From top to bottom and from left to right: several frames, potential fields, initial position and final position of the first model.

5 Conclusions

We have presented a combined variability approach to learn visual task and human gesture models by means of eigenspaces. We have presented the general gesture model and the set of perceptual processes which perform learning, tracking and recognition. Constraint design and learning are de-

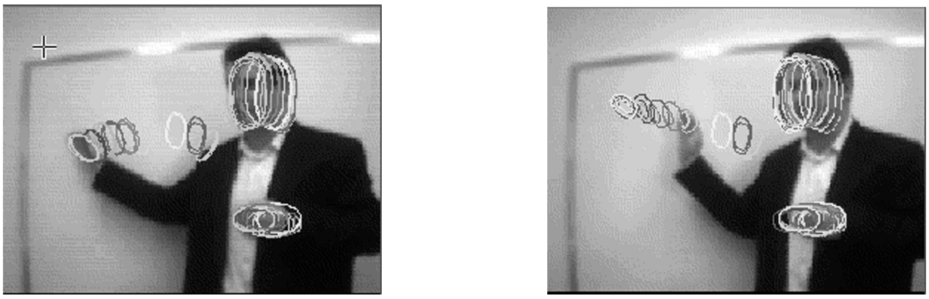

Fig. 5. Coupled Tracking. Left: results using the first gesture model. Right: result with the second model. The input sequence fits the first model.

tailed and practical tracking and recognition results are presented.

References

1. Baumberg, A., Hogg, D.: Learning Flexible Models in Image Sequences. European Conference on Computer Vision. (1994).
2. Black, M.J., Jepson, A.D.: EigenTracking: Robust Matching and Tracking of Articulated Objects Using a View-Based Representation. In Proc. ECCV-96, Cambridge 329-342.(1996).
3. Cédras, C., Shah, M.:Motion Based Recognition: A Survey. Tech-Rep. Department of Computer Science. Univ. of Central Florida. (1995).
4. Cootes,T.F., Taylor, C.J., Cooper, D.H., Graham J.: Trainable Method of Parametric Shape.Image and Vision Computing. **10** 289-294. (1992).
5. Cootes, T.F., Taylor, C.J.: Active Shape Models. Smart Snakes. Proc. British Machine Vision Conference. 266-275. (1992).
6. Cootes, T.F., Taylor, C.J., Lanitis, A., Cooper, D.H., Graham, J.: Building and Using Flexible Models Incorporating Grey-Level Information. In Proc. International Conference of Computer Vision. (1993).
7. Davis, J.W.: Appearance-Based Motion Recognition of Human Action. MIT Media Lab Tech Rep. 387. (1996).
8. Darrell, T.J., Essa, I.A., Pentland, A.P.: Task Specific Gesture Analysis in Real Time Using Interpolated Views. IEEE Trans. PAMI.**18** 1236-1242. (1996).
9. Escolano, F.: Plantillas Deformables Extendidas: Modelización Local Robusta y Caracterización Basada en Auto-Modelos para el Reconocimiento y Tracking de Estructuras Geométricas Activas. Tesis Doctoral. Universidad de Alicante (1997).
10. Escolano, F., Cazorla, M., Gallardo, D., Rizo, R.: Deformable Templates for Tracking and Analysis of Intravascular Ultrasound Sequences. EMMCVPR-97: International Workshop on Energy Minimization Methods for Computer Vision and Pattern Recognition. Lecture Notes in Computer Science N.1223. Springer Verlag 521-534.(1997).
11. Escolano, F., Cazorla, M., Gallardo, D., Llorens, F., Satorre R., Rizo, R.: Spatio-Temporal Deformable Templates for Gesture Tracking and Recognition. VII Conference of the Spanish Association of Artificial Intelligence (1997).

12. Fukunaga, K.: Introduction to Statistical Pattern Recognition. New York: Academic (1972).
13. Kirby, M., Weisser, F., Dangelmayr, G.: A model problem in the representation of digital image sequences. Patter Recognition. **26** 63-73. (1993).
14. Lanitis, A., Taylor, C.J., Cootes, T.F., Ahmed, T.: Automatic Interpretation of Human Faces and Hand Gestures Using Flexible Models. Tech-Rep. Department of Medical Biophysics. University of Manchester. (1994).
15. Moghaddam, B., Pentland, A.: Face Recognition using View-Based and Modular Eigenspaces. M.I.T. Technical Report No. 301. (1994)
16. Murase, H., Sakai, R.: Moving object recognition in eigenspace representation: gait analysis and lip reading. Patter Recognition Letters. **17**. 155-162. (1996).
17. Nastar, C., Ayache, N.: A New Physically Based Model for Efficient Tracking and Analysis of Deformations. In Proc. Geometric Reasoning for Perception and Action. Springer-Verlag (1993).
18. Turk, M., Pentland, A.: Eigenfaces for Recognition. Journal of Cognitive Neuroscience.**3** 71-89. (1991).
19. Zhu, S.C., Yuille, A.L.: FORMS: A Flexible Object Recognition and Modeling System. Int. Journal of Computer Vision. (1996)