Upgrading Eigenspace-based Prediction using Null Space and its Application to Path Prediction

Yuji Shinomura
Hiroshima University, Japan
shino@eml.hiroshima-u.ac.jp

Toru Tamaki
Hiroshima University, Japan
tamaki@ieee.org

Toshiyuki Amano
NAIST, Japan
amano@is.naist.jp

Kazufumi Kaneda
Hiroshima University, Japan
kin@hiroshima-u.ac.jp

Abstract

This paper proposes a method for an Eigenspace-based prediction of a vector with missing components by modifying a projection of conventional Eigenspace method, and demonstrates the application to the prediction of the path of a walking person. This modification is based on domain-specific knowledge of data, and a linear combination of vectors in the null space *of Eigenspace is added so that a cost function of smoothness of path is minimized. Some experimental results on actual paths are shown to demonstrate how the proposed method works.*

1   Introduction

It is useful to estimate or predict unknown future data from previously observed data in past or present not only for meteorology and economics, but also for computer vision. Generally, the AR model or Kalman filter are used to estimate time series of data. Although predicting gestures and tracking people also use similar methods, a prediction for such sequences is not so simple because usually their behavior cannot be captured by the Gaussian signal model. On the other hand, patterns of behavior and motion of people in daily life have few variations: same gesture has similar motion and a same person walks in similar paths in a same scene. Thus, scene-dependent information of time series in such applications can be learned as prior knowledge in advance.

Eigenspace approach has been widely used to learn such domain-specific information from samples. Fod et al.[4] and Yacoob et al.[10] used Eigenspace to recognize motion of a person, and Nakajima et al.[7] predicted spatially and temporally to recognize gestures by Eigenspace made from sample gestures. These methods use learning of Eigenspace

E with samples, and recognition and prediction are performed based on projection of a vector x onto Eigenspace spanned by several eigenvectors e_j:

$$\boldsymbol{a} = E^T \boldsymbol{x}, \quad E = [\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots]. \tag{1}$$

A problem of these Eigenspace-based methods is that they merely use a projection of a vector with all components: i.e., for a vector to be recognized or predicted, it should have the same dimensionality as the samples that were used to construct the Eigenspace. However, we have no components corresponding to future data for prediction, and occluded data for recognition. A simple solution is to just put 0 for such missing components in the vector:

$$\hat{\boldsymbol{a}} = E^T \hat{\boldsymbol{x}}, \quad \hat{\boldsymbol{x}} = [x_1, x_2, \ldots, x_p, 0, 0, \ldots]^T, \tag{2}$$

but its result is awful[6] and a reconstructed vector $E\hat{\boldsymbol{a}}$ is not similar to the original vector x.

One way to reconstruct a vector without using all pixels has been proposed by Leonardis et al.[6] to achieve a robust recognition when an object is occluded. Fidler et al.[3] utilized it to make LDA robust. Nakajima et al.[7] used a similar method for reconstruction and prediction, and Amano [2, 1] proposed methods to fill-in occluded regions. These methods are good for discrimination or recognition, but seem to fail to reconstruct or predict a vector with missing components[1] because characteristics of domain-specific data, such as smoothness or continuity, are ignored. These are summarized in section 2.

In this paper, we propose a new Eigenspace-based prediction method by modifying the conventional projection-based prediction with domain-specific knowledge of data,

---

[1] These missing components can be regarded as outliers, but robust subspace techniques such as a robust PCA proposed by De la Torre et al.[5] are not applicable because there are outliers not in learning samples but in a new test sample and also our case more than 50% components in the test sample are missing.

and demonstrate an application to predict the walking path of people. The modification uses *Null space*, the orthocomplement of Eigenspace, and a linear combination of vectors in the null space (*null vectors*) is added to the prediction so that a reconstructed vector with missing components (in our case, a person's walking path) satisfies some characteristic of data such as smoothness. Coefficients of the linear combinations are computed by the decent gradient method.

The organization of this paper is as follows. Eigenspace-based prediction is explained in Sec.2, then in Sec.3 we describe modification of the prediction with null vectors and estimation of linear combination of null vectors with the gradient decent method. Experimental results on actual paths are shown in Sec.4.

2 Eigenspace-based Prediction of a Path

This section introduces a prediction of a person's path based on projection onto Eigenspace.

2.1 Construction of Eigenspace with Sample Paths

In this paper, a path of a person is defined as a sequence of successive coordinates of the person over frames. Here we describe how to obtain a sequence of a path for learning.

First, regions in a frame where changes in intensity occur are extracted by using background subtraction. Then, the size of each region is used to reject regions other than people. The center of gravity of a region is used as a position of a person in the frame.

N paths are acquired for learning, then the paths are normalized in length that is defined as a sum of Euclidean distance between two successive coordinates. First the shortest path in the N paths is chosen. All paths are cut to the shortest length, then resampled so that all paths have the same length, M number of coordinates. Each i-th normalized path is represented by a vector $\boldsymbol{y}_i$ with $2M$ elements $\boldsymbol{p}_t$ as follows:

$$\boldsymbol{y}_i = (\boldsymbol{p}_1^T, \boldsymbol{p}_2^T, \ldots, \boldsymbol{p}_M^T)^T \in \mathbb{R}^{2M}, \quad (3)$$
$$\boldsymbol{p}_t = (p_{x_t}, p_{y_t})^T \in \mathbb{R}^2, \quad (4)$$

where $\boldsymbol{p}_t$ is a 2D vector representing t-th coordinates in a path.

Eigenspace E is constructed with the normalized N sample paths $\boldsymbol{y}_i$ that are centered by subtracting an average vector $\boldsymbol{m}$ ($= \frac{1}{N}\sum_{i=1}^N \boldsymbol{y}_i$) in advance, then eigenvectors $\boldsymbol{e}_i$ are computed:

$$E = [\boldsymbol{e}_1, \cdots, \boldsymbol{e}_N], \quad (5)$$
$$\boldsymbol{e}_i = (\boldsymbol{e}_{i1}^T, \boldsymbol{e}_{i2}^T, \ldots, \boldsymbol{e}_{iM}^T)^T \in \mathbb{R}^{2M}, \quad (6)$$

where E represents a matrix of Eigenspace spanned by the eigenvectors (or *Eigenpaths*) $\{\boldsymbol{e}_i\}$, and $\boldsymbol{e}_{it} \in \mathbb{R}^2$ corresponds to t-th 2D coordinates in $\boldsymbol{e}_i$.

2.2 Eigenspace-based Path Prediction

In prediction, a path of a new person is not fully traced, and there is no coordinates of the person in future. Suppose that a new person is tracked and the path is normalized to have s coordinates $\boldsymbol{p}'_1, \ldots, \boldsymbol{p}'_s$ as the same way for the learned paths.

$$\boldsymbol{y}' = (\boldsymbol{p}_1'^T, \ldots, \boldsymbol{p}_s'^T)^T \in \mathbb{R}^{2s}, \quad \text{where } s \leq M. \quad (7)$$

For unknown coordinates $\boldsymbol{p}'_{s+1}, \ldots, \boldsymbol{p}'_M$, we set them to zero $\boldsymbol{p}'_t = \boldsymbol{0} = (0,0)^T$, then an extended vector $\boldsymbol{y}''$ is obtained:

$$\boldsymbol{y}'' = (\boldsymbol{p}_1'^T, \ldots, \boldsymbol{p}_s'^T, \underbrace{\boldsymbol{0}^T, \ldots, \boldsymbol{0}^T}_{(M-s)})^T \quad (8)$$
$$= (\boldsymbol{y}'^T, \underbrace{0, \ldots, 0}_{2(M-s)})^T \in \mathbb{R}^{2M}. \quad (9)$$

In the framework of conventional Eigenspace methods, the observed vector $\boldsymbol{y}''$ is represented by a linear combination of the eigenvectors so that the following L-2 error norm is minimized [6] with respect to $\boldsymbol{a}$:

$$||\boldsymbol{y}'' - E\boldsymbol{a}||^2 = \left\| \boldsymbol{y}'' - \sum_j^N a_j \boldsymbol{e}_j \right\|^2 \quad (10)$$
$$= \sum_t^M \left\| \boldsymbol{p}'_t - \sum_j^N a_j \boldsymbol{e}_{jt} \right\|^2, \quad (11)$$

where $\boldsymbol{a} = (a_1, a_2, \ldots, a_N)^T$ is the coefficient of the linear combination. In our case, unknown coordinates are set to zero, so the norm is rewritten as:

$$||\boldsymbol{y}'' - E\boldsymbol{a}||^2 = \sum_{t=1}^s \left\| \boldsymbol{p}'_t - \sum_j^N a_j \boldsymbol{e}_{jt} \right\|^2$$
$$+ \sum_{t=s+1}^M \left\| \sum_j^N a_j \boldsymbol{e}_{jt} \right\|^2. \quad (12)$$

The second term in the above equation affects greatly the estimates of the coefficient $\boldsymbol{a}$. Instead, using only the first term and omitting the second term lead to a more appropriate estimate of the coefficient. This estimation is done by solving the following linear system [6, 3]:

$$E'^T E' \boldsymbol{a} = E'^T \boldsymbol{y}'', \quad (13)$$
$$E' = \text{diag}(\overbrace{1, \cdots, 1}^{2s}, \overbrace{0, \cdots, 0}^{2(M-s)}) E \quad (14)$$

where E' is a subspace of E spanned by truncated eigenvectors, but their basis are no longer orthogonal to each other. Note that $\mathrm{rank}(E'^T E') = N$ or $\det(E'^T E') \neq 0$ should be held so that the linear system doesn't become underdetermined. This means $2s > N$, hence the estimation can be done after several positions of a person are observed.

The reconstruction with the estimated coefficients $\boldsymbol{a}$ is as follows [2, 1, 7]:

$$\boldsymbol{y}^* = E\boldsymbol{a} = E(E'^T E')^{-1} E'^T \boldsymbol{y}''. \tag{15}$$

2.3 Modifying a Projection Outside of Eigenspace

The predicted path $\boldsymbol{y}^*$ is represented by a linear combination of eigenvectors $\boldsymbol{e}_i$,

$$\boldsymbol{y}^* = E\boldsymbol{a} = a_1\boldsymbol{e}_1 + a_2\boldsymbol{e}_2 + \cdots + a_N\boldsymbol{e}_N = \sum_{i=1}^{N} a_i\boldsymbol{e}_i. \tag{16}$$

However, Eq.(15) shows us that $\boldsymbol{y}^*$ is a projection of $\boldsymbol{y}''$ onto a subspace spanned by non-orthonormal vectors[8, 3], in this case not E but the truncated subspace E'. Therefore, there is no reason to believe that the projection represents the original data well because the truncation of the Eigenspace depends not on principal components corresponding to small eigenvalues (usually referred as dimensionality reduction) but just the length of observation. Also, this projection does not take into account the characteristics of a person's walking path, and the estimated path $\boldsymbol{y}^*$ results in something different from a real path.

In this paper, we propose the use of the orthocomplement of the Eigenspace, denoted as $E^\perp$, where $\mathbb{R}^{2M} = E + E^\perp$. All vectors in $E^\perp$ are orthogonal to any vectors in E, and vice versa: e.g., $\forall \boldsymbol{\ell} \in E^\perp \Rightarrow E\boldsymbol{\ell} = 0$. Therefore, we call $E^\perp$ as the *null space* of E, and a vector in the null space is called a *null vector*. By using null vectors in the null space, a path is represented as follows:

$$\widetilde{\boldsymbol{y}} = \boldsymbol{y}^* + \sum_{k} b_k\boldsymbol{\ell}_k = \sum_{i=1}^{N} a_i\boldsymbol{e}_i + \sum_{k} b_k\boldsymbol{\ell}_k \tag{17}$$

Estimated path $\boldsymbol{y}^*$ in Eq.(15) is identical to the equation above when coefficients b_k for null vectors in the second term are zero.

The concept of the proposed method is that domain-specific knowledge discarded by the conventional projection can be found in the null space if we can find the appropriate coefficients b_k for the null vectors $\boldsymbol{\ell}_k$. This topic is described in the next section. It should be noted that the projection of $\widetilde{\boldsymbol{y}}$ onto E is still $\boldsymbol{y}^*$.

3 Null Vector Modifications

The proposed method shown in this section adds null vectors to the projected path $\boldsymbol{y}^*$ so that the modified path $\widetilde{\boldsymbol{y}}$ looks like a person's walking path. In this paper, we make an assumption that *a person walks toward a destination, and does not turn suddenly, and the path is smooth and does not have a sharp curve*. Here we introduce a cost function of smoothness of a path that has never been used by conventional Eigenspace-based estimations.

First, we assume that K null vectors $\boldsymbol{\ell}_k = (\boldsymbol{\ell}_{k1}^T, \boldsymbol{\ell}_{k2}^T, \ldots, \boldsymbol{\ell}_{kM}^T)^T \in E^\perp$ are given. Then the linear representation of the modified path $\widetilde{\boldsymbol{y}}$ is:

$$\begin{aligned}
\widetilde{\boldsymbol{y}} &= \sum_{i=1}^{N} a_i\boldsymbol{e}_i + \sum_{k=1}^{K} b_k\boldsymbol{\ell}_k \\
&= (\widetilde{\boldsymbol{p}}_1^T, \widetilde{\boldsymbol{p}}_2^T, \ldots, \widetilde{\boldsymbol{p}}_M^T)^T, \tag{18} \\
\widetilde{\boldsymbol{p}}_t &= \sum_{i=1}^{N} a_i\boldsymbol{e}_{it} + \sum_{k=1}^{K} b_k\boldsymbol{\ell}_{kt}. \tag{19}
\end{aligned}$$

Let $\boldsymbol{u}_t$ be a vector defined by two successive[2] coordinates $\widetilde{\boldsymbol{p}}_t, \widetilde{\boldsymbol{p}}_{t+1}$, and θ_t be an angle subtended by $\boldsymbol{u}_t$ and $\boldsymbol{u}_{t+1}$:

$$\boldsymbol{u}_t = \widetilde{\boldsymbol{p}}_{t+1} - \widetilde{\boldsymbol{p}}_t, \tag{20}$$

$$\cos\theta_t = \frac{\boldsymbol{u}_t^T \boldsymbol{u}_{t+1}}{||\boldsymbol{u}_t||\,||\boldsymbol{u}_{t+1}||}, \quad 1 \leq t \leq M - 2. \tag{21}$$

Next, we define a cost function J so that the smaller the angle θ_t is the smoother the path is:

$$J = \sum_{t=1}^{M-2} \cos^\alpha \theta_t, \qquad \alpha = 1, 3, 5, \ldots \tag{22}$$

The steepest gradient method is used to maximize the cost function for coefficients of the null vectors b_k ($k = 1, \ldots, K$) as $b_k \leftarrow b_k + \frac{\partial J}{\partial b_k}$, and all b_k are initialized to 0. A stopping condition is $\max_k \left| \frac{\partial J}{\partial b_k} \right| < 10^{-5}$. The Jacobian of J comprises $\boldsymbol{u}_t$ and $\boldsymbol{\ell}_t$ (omit detail).

In the discussion above, we assume that the null vectors are given. However, there are no established methods to get null vectors. Also there are a lot of variations to choose null vectors from the null space. For example, assume that there are 13 paths comprised of 250 coordinates given as samples. The dimensionality of the Eigenspace is up to 13, however, the null space has $500 - 13 = 487$ dimensions. Usually the number of samples is much fewer than the number of coordinates in a path. Therefore it is difficult to find the most appropriate null vector to modify the predicted path.
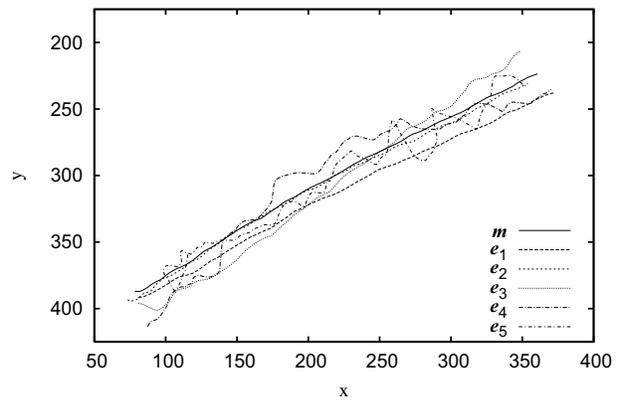
---

[2]Of course, we can two coordinates distant from each other $\widetilde{\boldsymbol{p}}_t$ and $\widetilde{\boldsymbol{p}}_{t+k}$. In this case, the sum of k-curvature (see, for example, [9]) over the path is minimized.
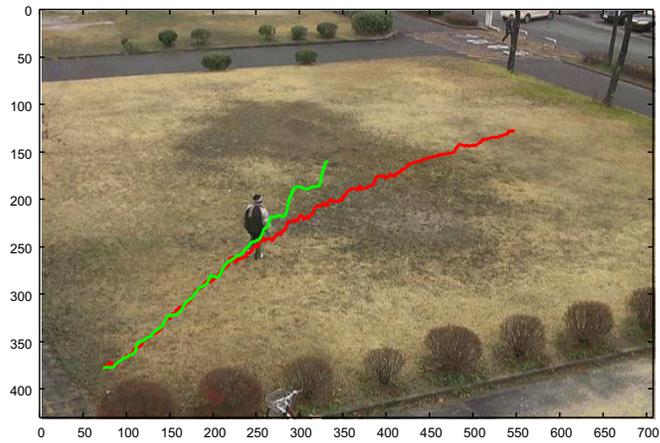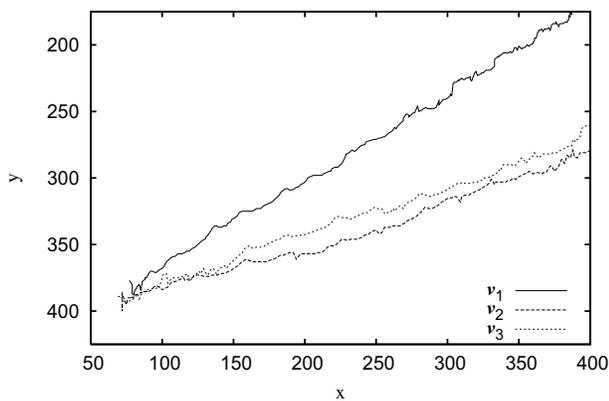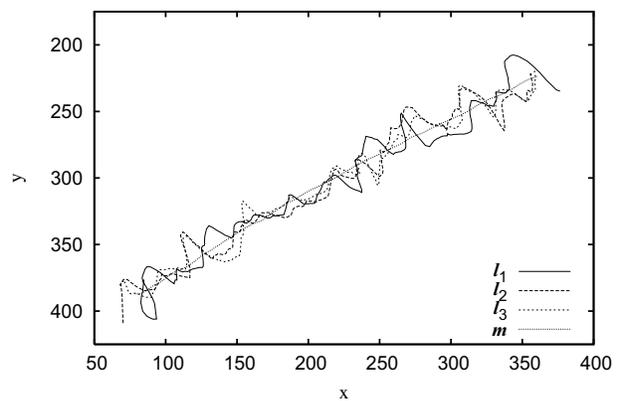
(a)
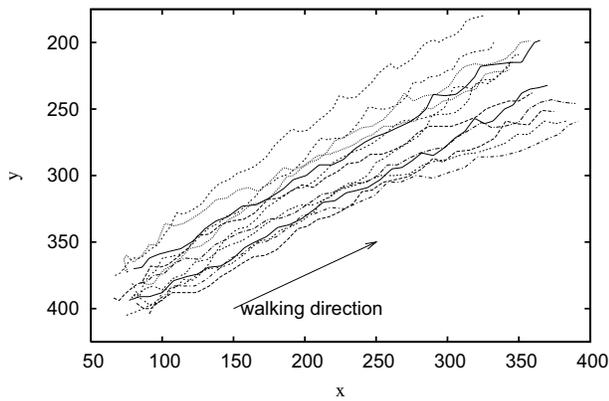
(b)

(c)

(d)

(e)

Figure 1. A frame of video and paths used in the experiments. (a) Predicted path y^* (green) and actual path y (red). (b) 13 sample paths $y_1, \ldots, y_{13}$, (c) 5 eigenvectors $e_1, \ldots, e_5$, (d) 3 samples used for null vectors $v_1, \ldots, v_3$ and (e) 3 null vectors $\ell_1, \ldots, \ell_3$. Note that e_j and ℓ_k are scaled properly for visualization.

Table 1. Initial and converged values of the cost function J with a null vector ℓ_1 for different α.

α	1	3	5	7	9
J (init.)	234.56	215.23	201.26	190.71	181.87
J (conv.)	235.20	217.95	206.06	196.84	189.18
b_1	-15.14	-27.36	-34.18	-36.96	-38.33

Table 2. Comparison of the number of the null vector.

	ℓ_1	ℓ_2	ℓ_3	$\sum b_k \ell_k$
K	1	1	1	3
J	235.20	234.67	239.88	240.727
b_1	-39.73	0	0	-4.80
b_2	0	-5.90	0	14.07
b_3	0	0	-34.54	-42.62

In this paper, null vectors are obtained from paths other than sample paths. In general, the dimension is so high that the new paths probably do not lay on the Eigenspace spanned by the sample paths. The null vectors ℓ_k in the null space are made from the new vectors v_k by using the Gram-Schmidt orthonormalization:

$$\ell'_k = v_k - \sum_{i=1}^{N}(v_k^T e_i)e_i - \sum_{j=1}^{k-1}(v_k^T \ell_j)\ell_j, \quad (23)$$

$$\ell_k = \frac{\ell'_k}{|\ell'_k|}. \quad (24)$$

We may add the new paths for null vectors to the learning sample paths to construct the Eigenspace instead to make the null space. This way seems to make use of the information of the new paths for better prediction, however, all information of learning samples are truncated by eq.(14), then no way to retrieve information corresponding to the missing components in the new paths.

4   Experimental Results

We implemented the proposed method, and evaluated using real image sequences of $714{\times}480$ in size. In the experiment, a video camera was fixed to a tripod, and movies were recorded as MPEG files, then 17 paths were obtained by off-line processing. People walked from the bottom left to the top right of the frame (Fig.1(a)). 13 paths were used as samples to make an Eigenspace (Fig.1(b)), and another three paths were used for null vectors (Fig.1(d)(e)). The remaining path is used for prediction (Fig.2). When learning, each path was cut so that it consisted of 350 coordinates, then 50 points are sparsely downsampled with linear interpolation for noise reduction. Finally $M = 250$ coordinates are resampled for a path. When predicting, a person is tracked and the path was normalized at each frame.

Predicted paths y^* with $N = 13$ are shown in Fig.2(a) for several different positions p'_s represented by $\bigcirc$. The prediction near to the start position (for small s) deviated largely from the actual path. As s increases, the prediction becomes similar to the actual path.

Next, Fig.2(b) shows the modification by a null vector ℓ_1. The estimated path y^* was predicted at $s = 100$.

Actually the modification is slight, but $\widetilde{y}$ is indeed more smoother than y^*. Table.1 shows values of the cost function and estimated coefficient b_1 when α changes. Although b_1 differs for different α, this variation is so small and does not affect the shape of the path because the null vector ℓ_1 has 500 elements but its norm is normalized to 1. Therefore, the choise of α is trivial and we set $\alpha = 1$ for all experiments.

Fig.3(a) illustrates results of modification by each null vector. Fig.3(b) shows the result by using 3 null vectors at the same time, and Table.2 shows the estimated parameters. Although the modified path depends on which path is used, the difference is small.

Another experiment is shown in Fig.4. Fig.4(a) shows 30 sample paths used to construct Eigenspace. Unlike the previous experiment, the walking path curves twice and looks like the S letter. Predicted and modified paths of a new path are shown in Fig.4(b) for different positions. This result shows that the proposed method is applicable to curved complex path in which prior knowledge is effectively used.

5   Conclusion

In this paper, we proposed a method for predicting a vector with missing components based on Eigenspace with null space modifications. We applied the method to paths of walking people in a real sequence, and demonstrated in the limited experiments how the proposed method works. There are many things to be considered, such as the number of the null vectors, the way to obtain the null vectors, the choice of other cost functions that represent domain-specific knowledge. Also futher experiments should be done. Nevertheless, the concept of the proposed method — to explore *out of the subspace* spanned by samples based on a prior knowledge — can be applicable to any other subspace recognition methods. We will investigate the possibility in other pattern recognition problems.
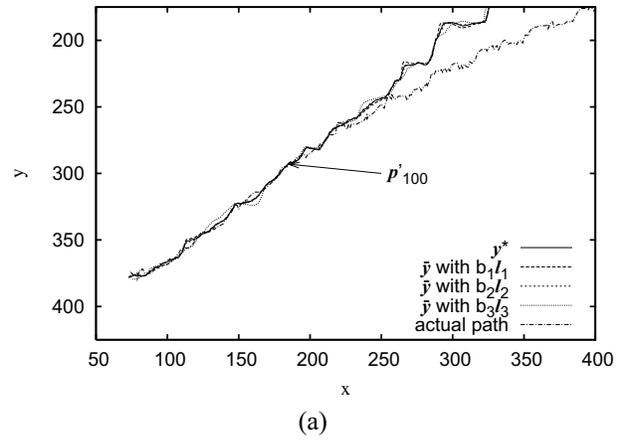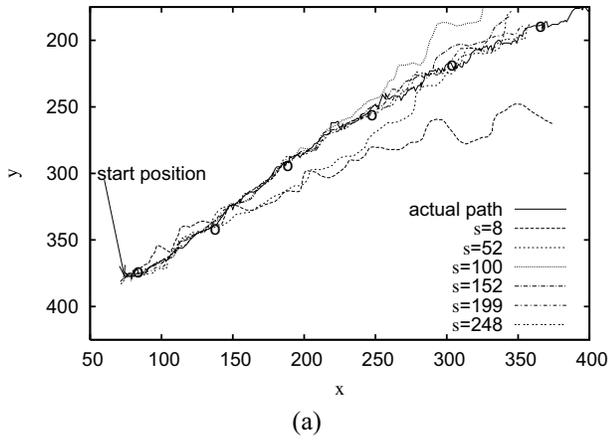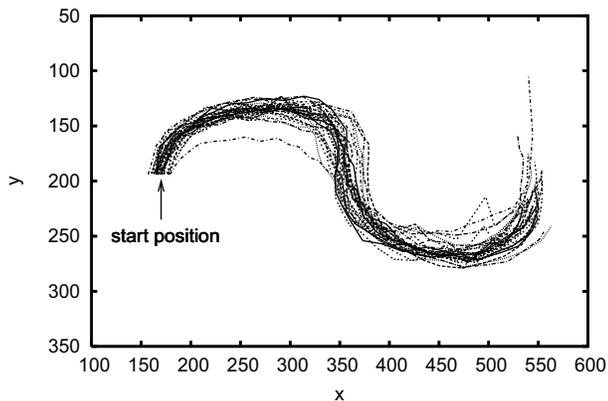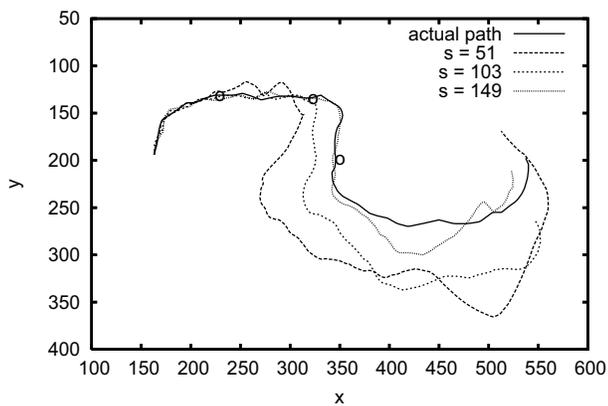
(a)

(a)

(b)

(b)
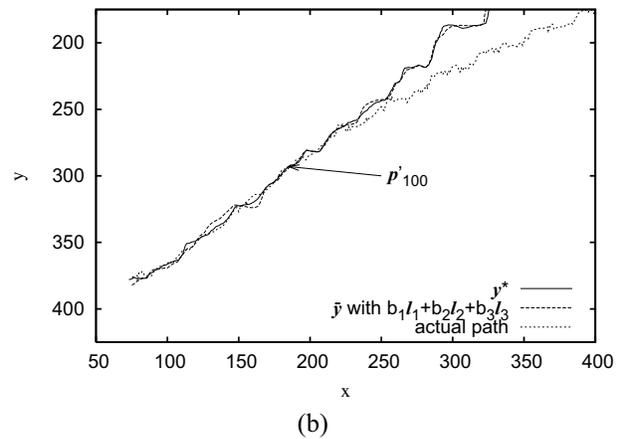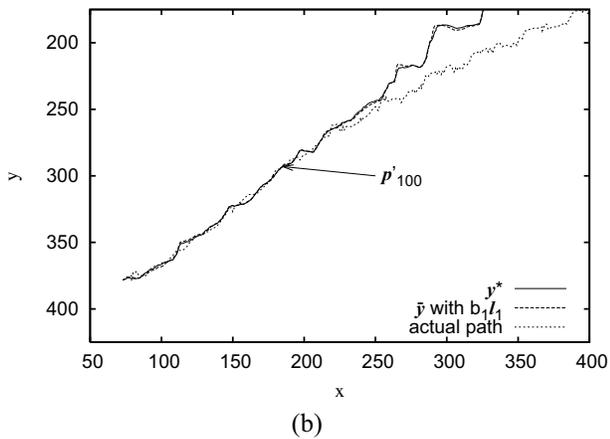
Figure 2. (a) Predicted paths y^* for different position s. (b) Predicted path y^* (solid) when $s = 100$, and modified path $\widetilde{y}$ (dashed).

Figure 3. (a) Modified path $\widetilde{y}$ using each null vector. (b) Modified path $\widetilde{y}$ using 3 null vectors.

[6] A. Leonardis and H. Bischof. Robust recognition using eigenimages. *Computer Vision and Image Understanding*, 78(1):99–118, 2000.

[7] M. Nakajima, S. Uchida, A. Mori, R. Kurazume, R. Taniguchi, T. Hasegawa, and H. Sakoe. Motion prediction based on eigen-gestures. *Proc. of the 1st First Korea-Japan Joint Workshop on Pattern Recognition (KJPR2006)*, pages 61–66, 2006.

[8] E. Oja. *Subspace Methods of Pattern Recognition*. Research Studies Press, 1983.

[9] Y. Shirai. *Three-Dimensional Computer Vision*. Springer-Verlag, 1987.

[10] Y. Yacoob and M. J. Black. Parameterized modeling and recognition of activities. *Computer Vision and Image Understanding*, 73(2):232–247, 1999.

(a)

(b)

Figure 4. (a) Learned 30 paths. $N = 30, M = 300$**. (b) Predicted and modified path.**

References

[1] T. Amano. Image interpolation by high dimensional projection based on subspace method. *Proc. of ICPR2004*, 4:665–668, 2004.

[2] T. Amano and Y. Sato. Image interpolation using BPLP method on the eigenspace. *Systems and Computers in Japan*, 38(1):457–465, 2007.

[3] S. Fidler, D. Skocaj, and A. Leonardis. Combining reconstructive and discriminative subspace methods for robust classification and regression by subsampling. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(3):337–350, 2006.

[4] A. Fod, M. J. Matarić, and O. C. Jenkins. Automated derivation of primitives for movement classification. *Autonomous Robots*, 12(1):39–54, 2002.

[5] F. D. la Torre and M. J. Black. A framework for robust subspace learning. *International Journal of Computer Vision*, 54:117–142, 2003.