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

ACCV2007 Workshop on Subspace 2007
19, Nov. .2007 at the University of Tokyo, Tokyo Japan

Yuji Shinomura — HIROSHIMA UNIVERSITY
Toru Tamaki — HIROSHIMA UNIVERSITY
Toshiyuki Amano — NAIST
Kazufumi Kaneda — HIROSHIMA UNIVERSITY

Background

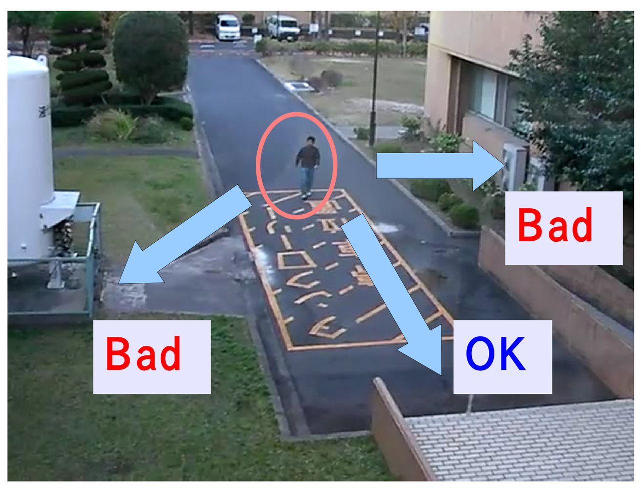
Surveillance camera system

Current : Tracking

Next step: Judgment of suspicious person

Judge

Future : Walking path prediction

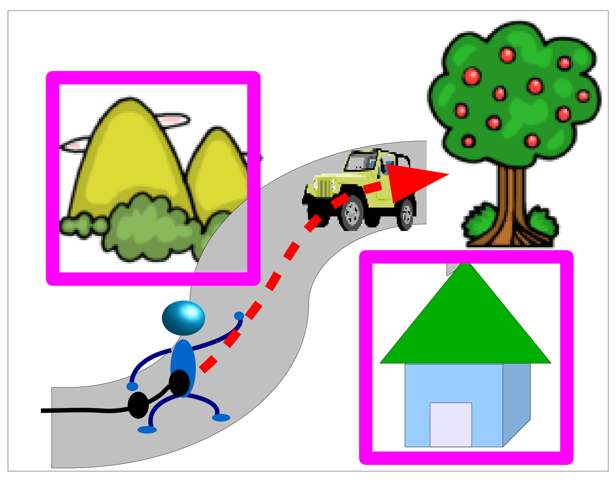


Path prediction methods

- Kalman Filter
- Autoregressive (AR) model
- Eigenspace-based (Yamamoto 2004)

Walking path condition

- Not simple
- Depend on walking environment (ex. Load, buildings, entrance, etc)

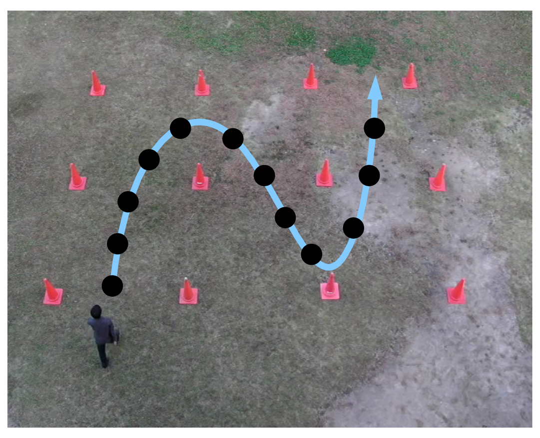


Eigenspace-based Prediction

Walking path

- a sequence of successive coordinates of the person over frames, and each position given by background subtraction

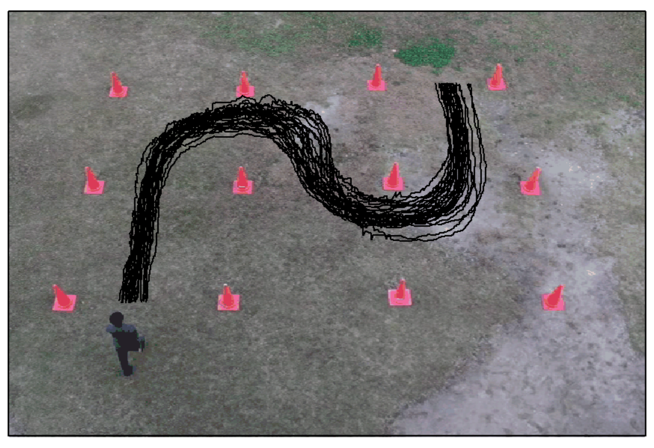
$\boldsymbol{p}_t = (p_{x_t}, \ p_{y_t}) \subseteq \mathbb{R}^2 \qquad [\boldsymbol{p}_1^T, \boldsymbol{p}_2^T, \ldots]^T$



Learning

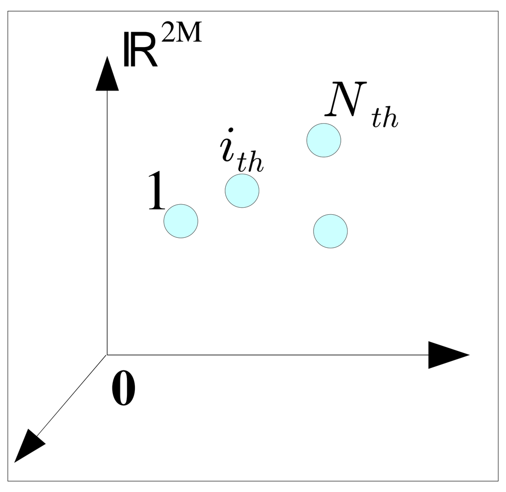
1. Learning N sample path

- Different sample paths have different number of frames



2. Path Normalization

- Different sample paths are normalized and coordinated with the number of $2M$ coordinates

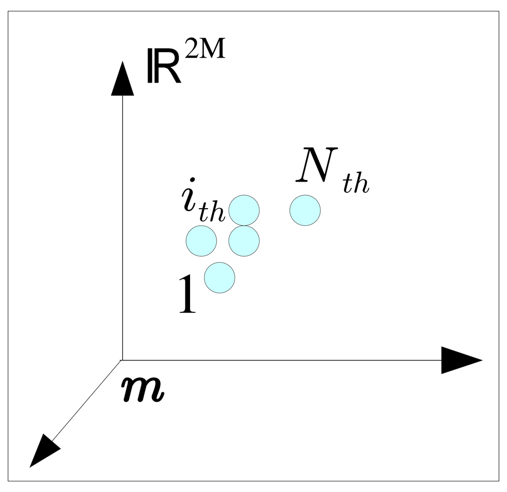


$\boldsymbol{y}_i = [\boldsymbol{p}_{i1}^T, \ \boldsymbol{p}_{i2}^T, \ \ldots, \ \boldsymbol{p}_{iM}^T]^T \subseteq \mathbb{R}^{2M}$

$\boldsymbol{y}_i - \boldsymbol{m}$

3. Average vector subtraction

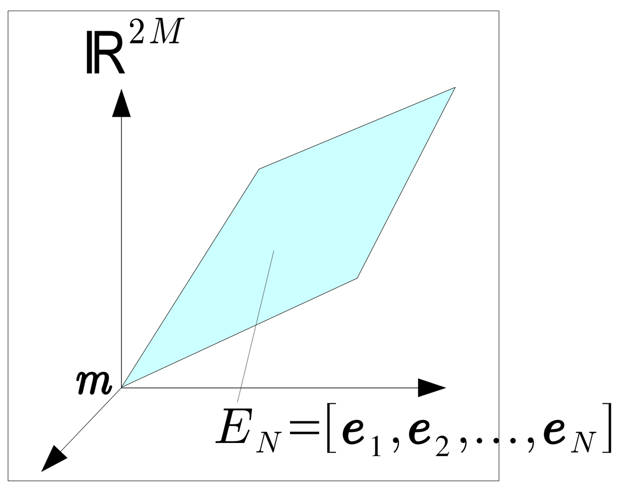
- Normalized N sample path are centered by subtracting an average vector $\boldsymbol{m}$



$\boldsymbol{m} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{y}_i$

4. Making Eigenspace

- Singular value decomposition computes eigenvectors e_i



$\boldsymbol{e}_i = [e_{i1}^T, e_{i2}^T, \ldots, e_{iM}^T]^T \subseteq \mathbb{R}^{2M}$

Prediction

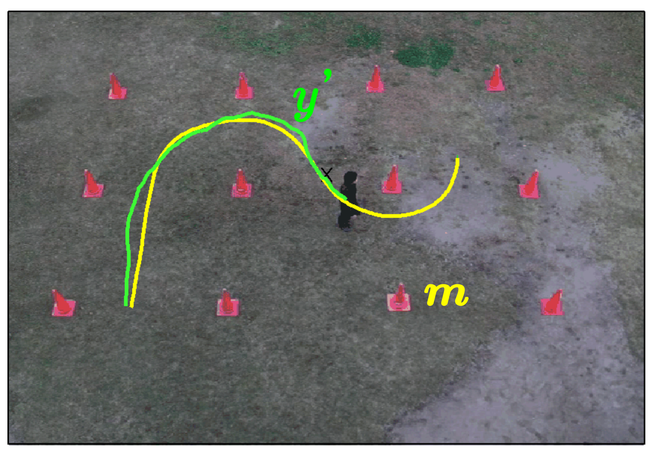
5. Tracking path $\boldsymbol{y}'$

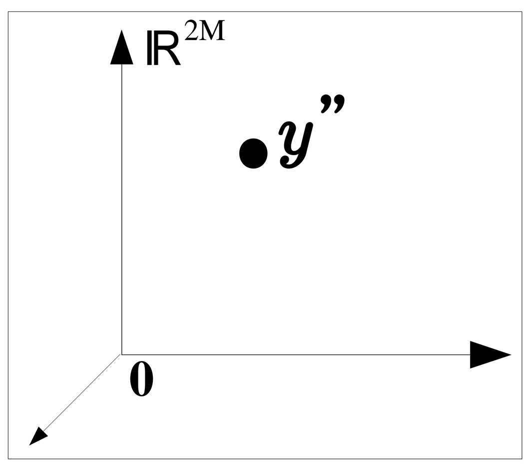
- Person is tracked at sth frame

$\boldsymbol{y}' = [\boldsymbol{p}_1^T, \boldsymbol{p}_2^T, \ldots, \boldsymbol{p}_s^T]^T \subseteq \mathbb{R}^{2s}$

6. Compensation

- Coordinates of $2(M-s)$ dimension are compensated by average vector $\boldsymbol{m}$
- Represent on $\mathbb{R}^{2M}$





$\boldsymbol{y}'' = [\boldsymbol{p}_1^T, \boldsymbol{p}_2^T, \ldots, \boldsymbol{p}_s^T, \boldsymbol{m}_{s+1}, \ldots, \boldsymbol{m}_M]^T \subseteq \mathbb{R}^{2M}$

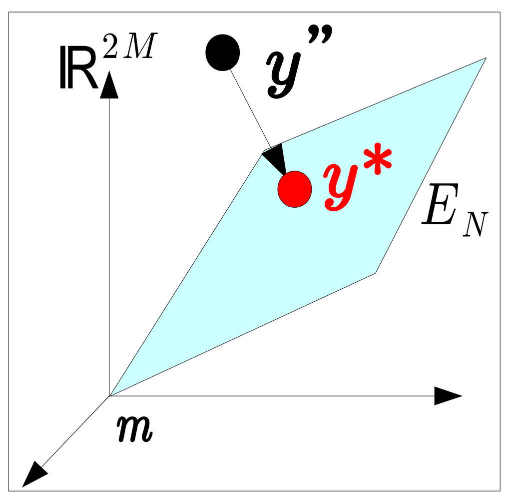
$\boldsymbol{y}'' - \boldsymbol{m}$

7. Projection onto Eigenspace

- Linear combination of eigenvectors

$$\boldsymbol{y}* = E\,\boldsymbol{a} = E(E'^T E')^{-1} E'^T \boldsymbol{y}'' = \sum_{i=1}^{N} a_i \boldsymbol{e}_i$$

$$E' = diag(\overbrace{1,\ldots,1}^{2s}, \overbrace{0,\ldots,0}^{2(M-s)})\,E$$



$\boldsymbol{y}* + \boldsymbol{m}$

8. Inverse projection

- Add average vector $\boldsymbol{m}$ to $\boldsymbol{y}*$

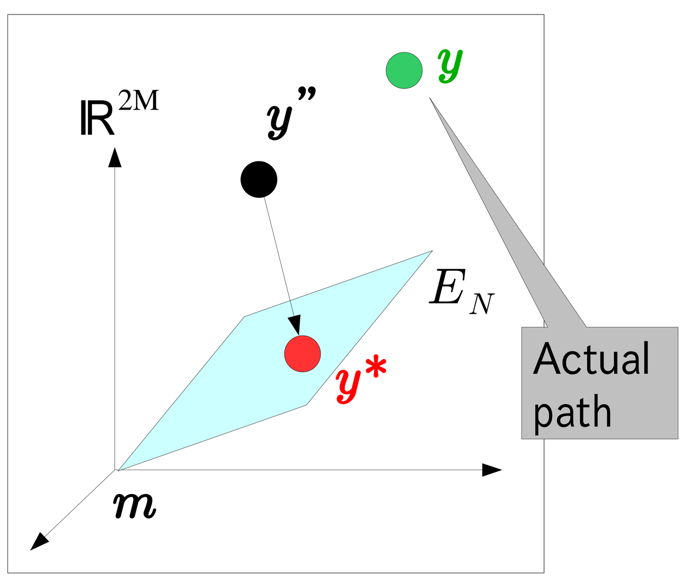
Problem & Objective

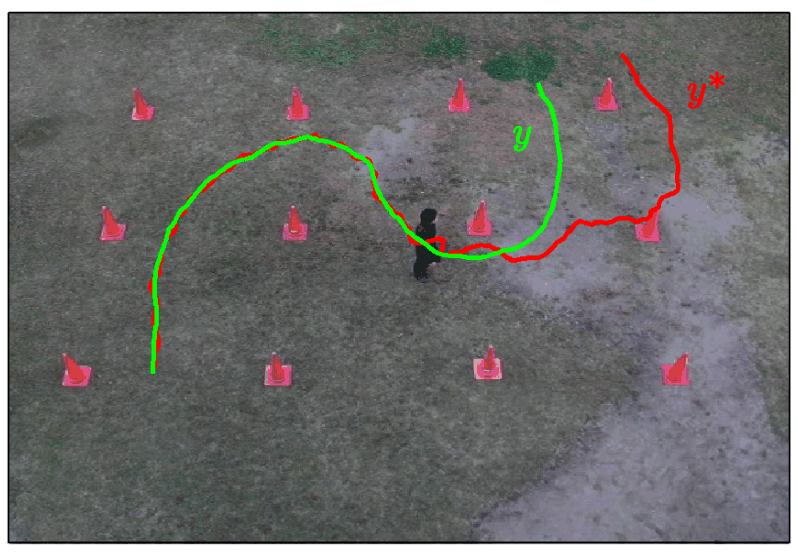
Problem

- Prediction is not correspond to actual path

Cause

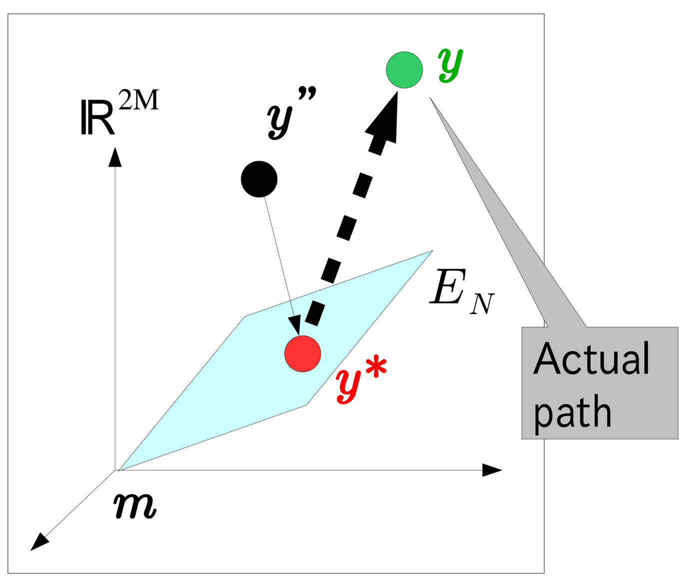
- Rack of eigenvectors in $(2M\text{-}N)$ Dimension





Objective

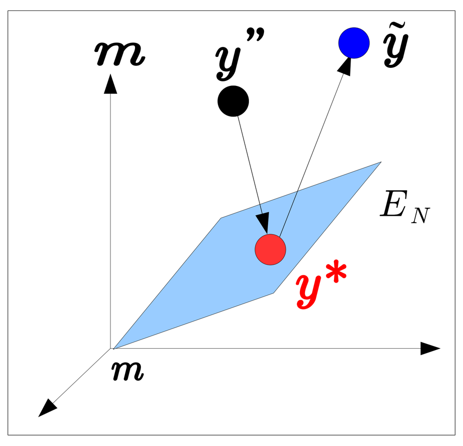
- Improvement of prediction result



Modifying a Projection using null vector in null space

IDEA: Use the orthocomplement of the Eigenspace

$$\tilde{y} = \underbrace{\sum_{i=1}^{N} a_i \boldsymbol{e}_i}_{y^*} + \underbrace{\sum_{k=1}^{K} b_k \boldsymbol{\ell}_k}_{\text{Modified part}}$$



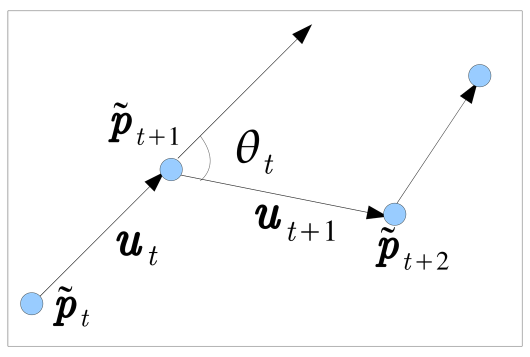
What is it needed to use the orthocomplement of the Eigenspace?

- $\boldsymbol{\ell}_k$: Null vector
- b_k : Coefficient of null vector

How to get coefficient of null vector b_k

Assumption

– Walking path is smooth



Cost function

– Consider the degree of smooth path

– Angle subtended by $\boldsymbol{u}_t$ and $\boldsymbol{u}_{t+1}$

$$maximize : J = \sum_{t=1}^{M-2} \cos^{\alpha}\theta_t \quad (\alpha = 1,3,5,\ldots)$$

$\cos\theta_t$ can be calculated easily as follows:

$$\cos\theta_t = \frac{\boldsymbol{u}_t^T \boldsymbol{u}_{t+1}}{\|\boldsymbol{u}_t\|\|\boldsymbol{u}_{t+1}\|}$$

Finally, the Jacobian of J comprises $\boldsymbol{u}_t$ and $\boldsymbol{\ell}_{kt}$

Iteration

- the steepest gradient method updates b_k , and make modificated path $\tilde{y}$

$$b_k \leftarrow b_k + \frac{\partial J}{\partial b_k}$$

(k: the number of null vector)

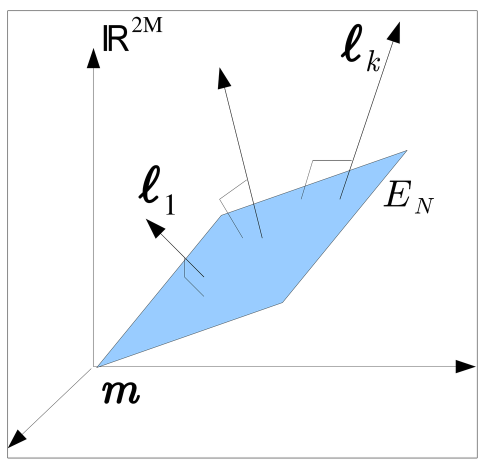
- A stopping condition

$$\max_k \left| \frac{\partial J}{\partial b_k} \right| < 10^{-5}$$

Null vector $\boldsymbol{\ell}_k$

– orthogonal vector of Eigenspace

– Null space $E^{\perp}$ consists of null vectors



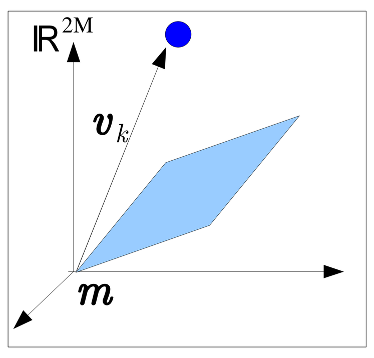
How to get Null vector $\boldsymbol{\ell}_k$

1. Learning new path that is not the same path used in making Eigenspace

ex)

- Obtaining new walking path
- Making new path from smoothing learning sample path

2. Subtraction of average vector



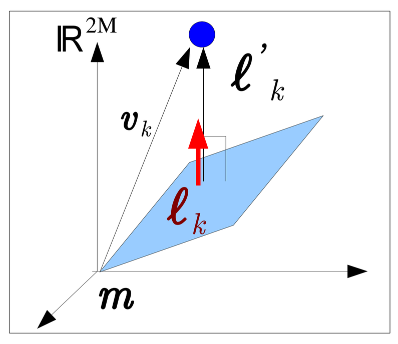
3. Gram-Schmidt orthonormalization

- Making orthognal vector of Eigenspace and other null vectors

$$\boldsymbol{\ell}'_k = \boldsymbol{v}_k - \sum_{i=1}^{N} (\boldsymbol{v}^T{}_k \boldsymbol{e}_i)\boldsymbol{e}_i - \sum_{j=1}^{k-1} (\boldsymbol{v}^T{}_k \boldsymbol{\ell}_j)\boldsymbol{\ell}_j$$

- Normalization

$$\boldsymbol{\ell}_k = \frac{\boldsymbol{\ell}'_k}{\|\boldsymbol{\ell}'_k\|}$$

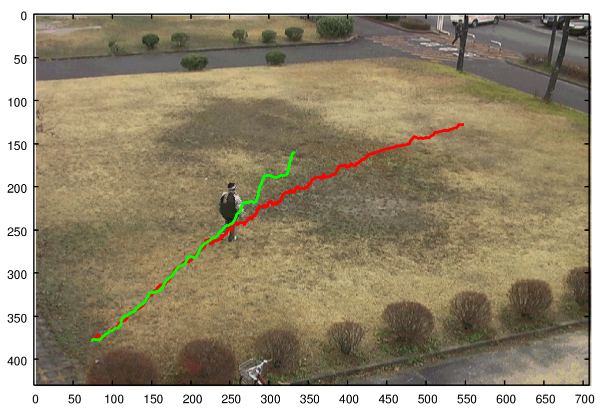


Experimental Results

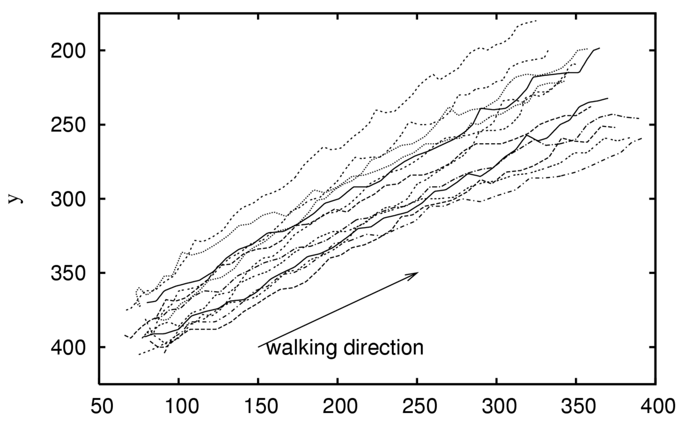
Case 1:

Learning

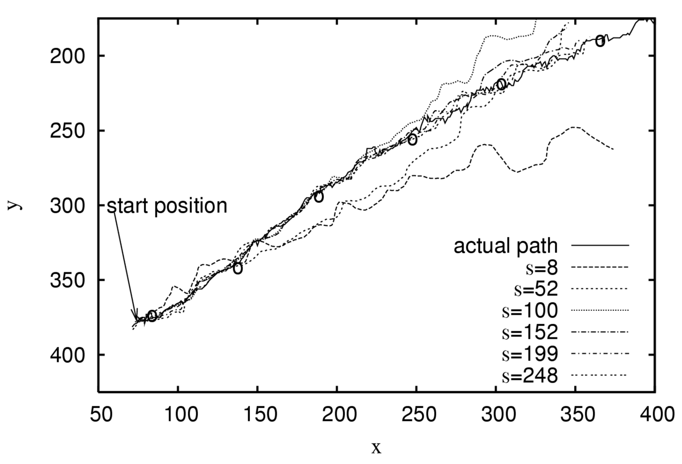
- Sample path : 13
- Downsampling: 50(plots)
- Resampling: 250(plots)



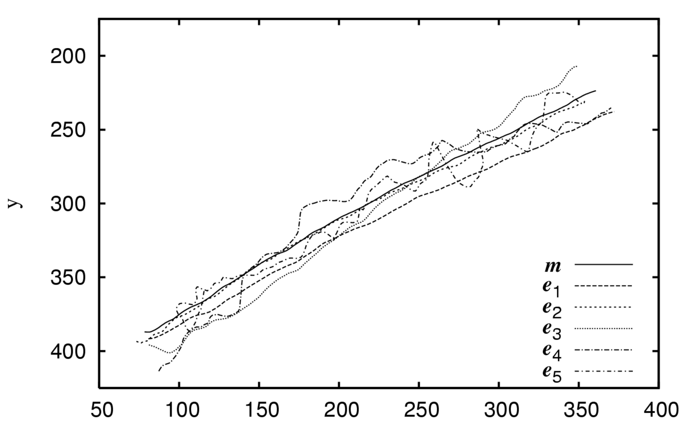
Prediction results.



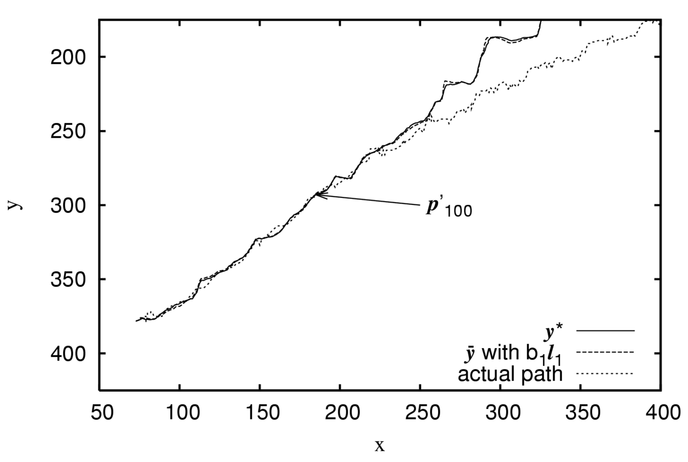
Learning 30 sample path.



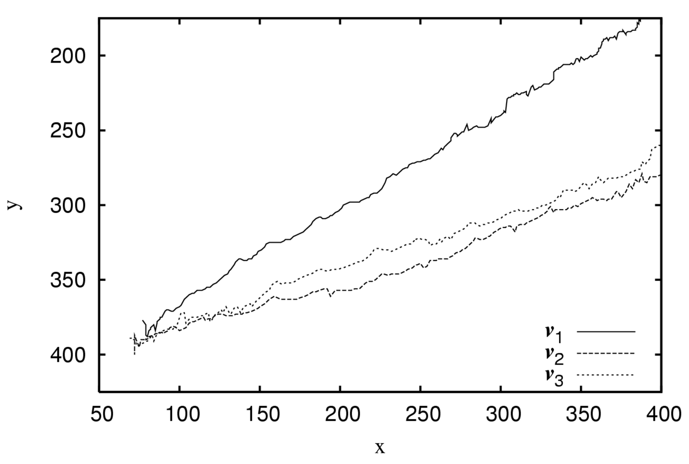
Prediction results. s=8,52,100,152,198,248.



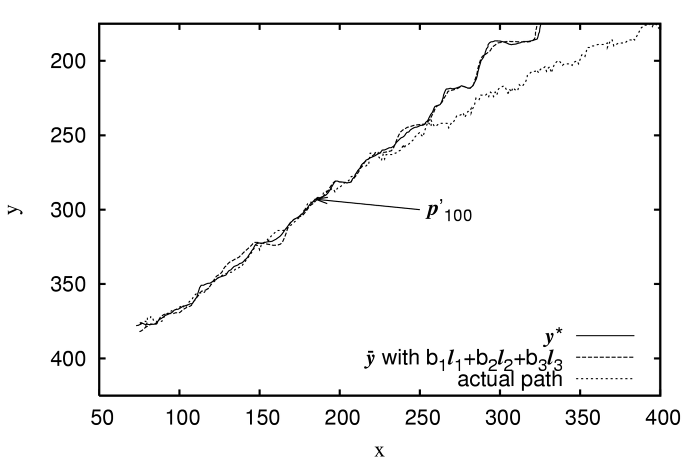
Eigenvectors .Each eigenvector is multiplied same number and is added average vector $\boldsymbol{m}$.



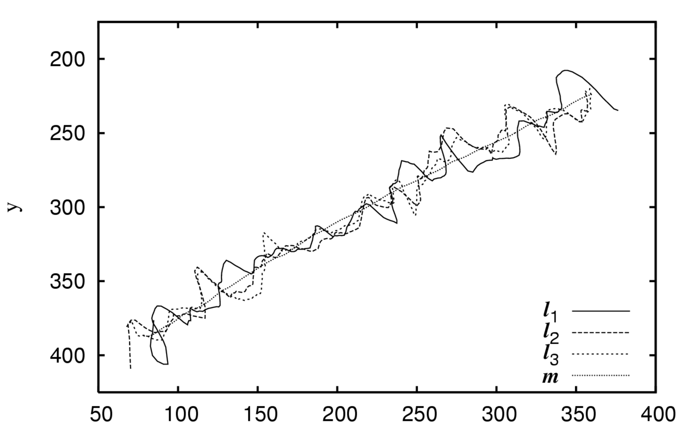
Modification result using 1 null vector.



New path for making null vector .

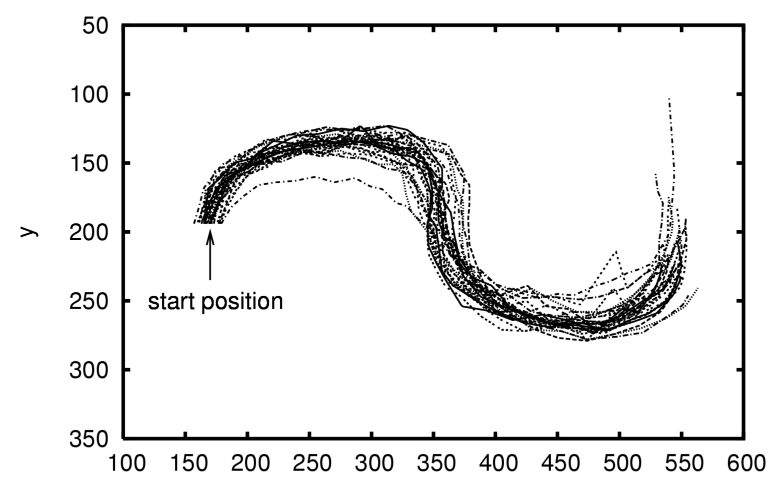


Modification result using 3 null vector.

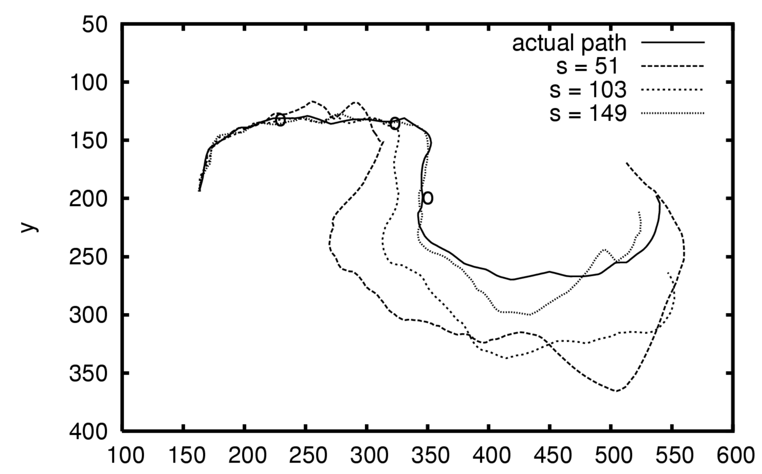


3 null vectors .Each null vector is multiplied same number and is added average vector $\boldsymbol{m}$.

Case 2:



Learning 30 sample path. These path are normalized as the same lengh.



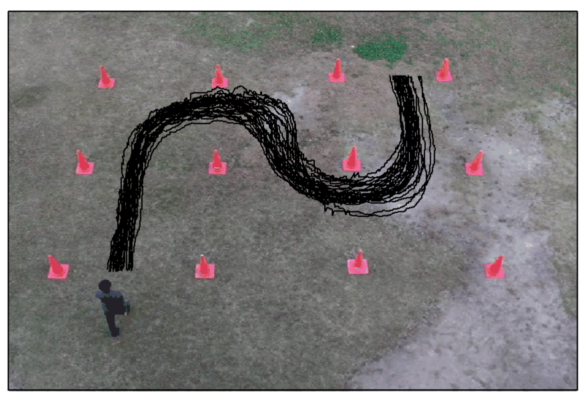
Modification result at 51, 103, 149 th coordinates.

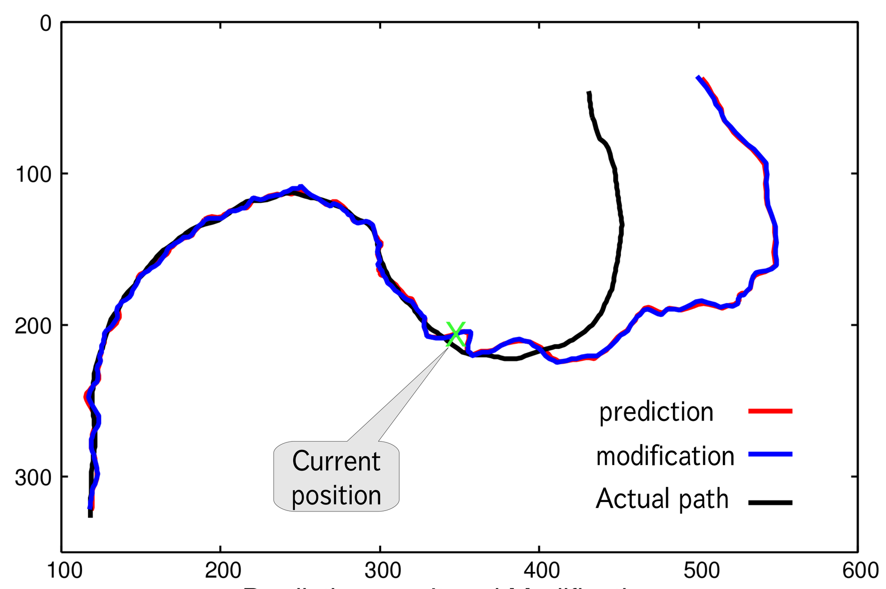
Case 3:

Learning

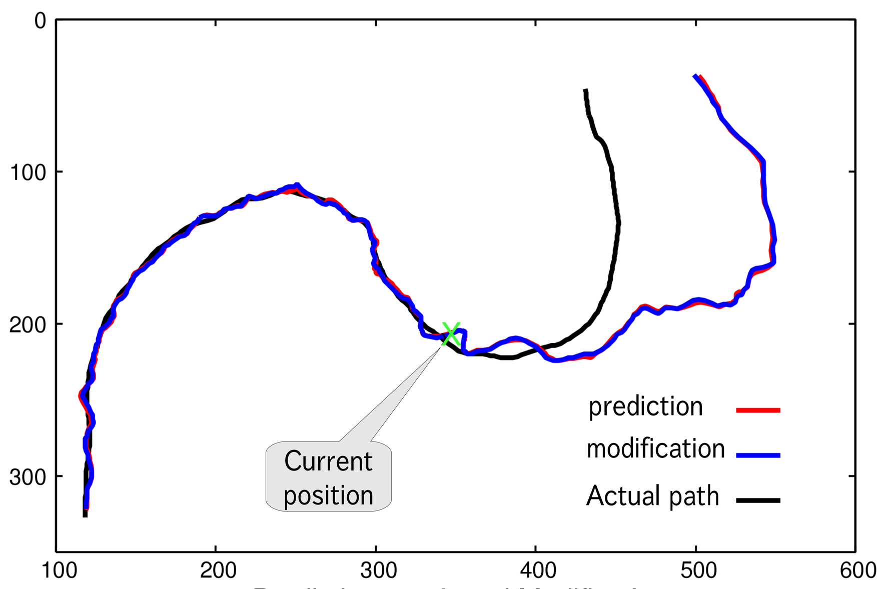
- Sample path : 30
- Downsampling: 50(plots)
- Resampling: 300(plots)

Modifying 1 null vector and 3 null vector





Prediction result and Modification result using 1 null vector.



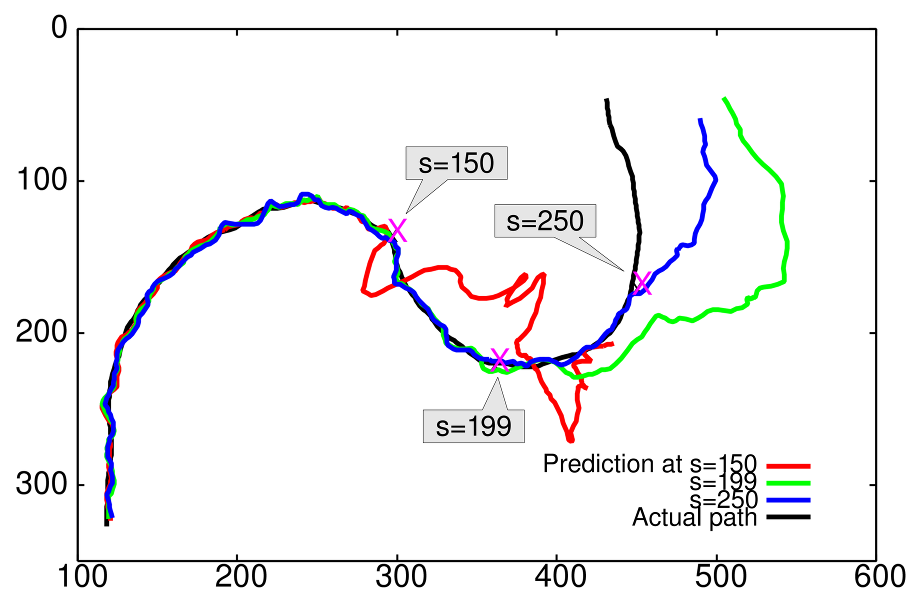
Prediction result and Modification result using 3 null vector.

Table: Results of Iteration using 1 null vector

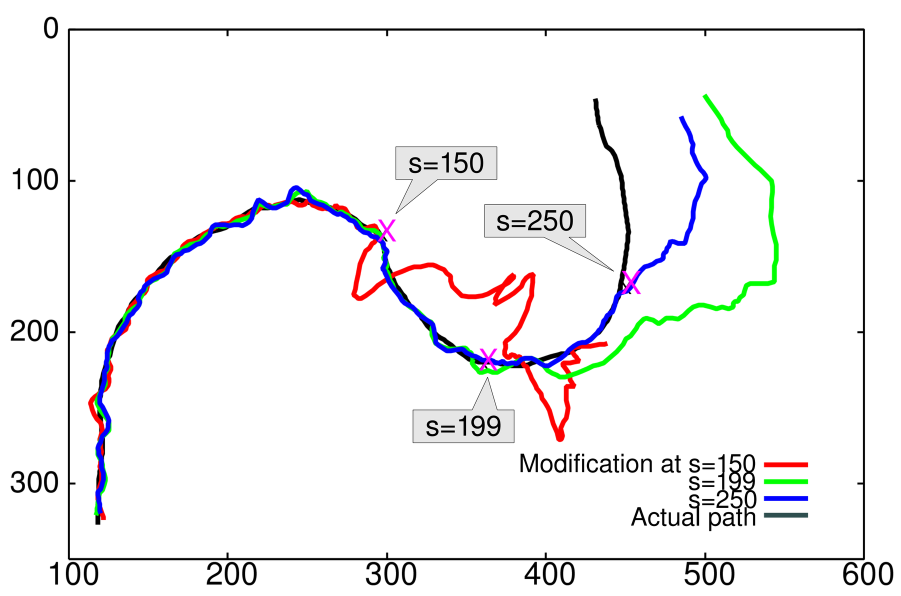
	Initial	After
Cost function: J	273.45	275.52
$\frac{\partial J}{\partial b_2}$	-0.2	-9.95E-06
Coefficient : b_2	0	-22.67

Table: Results of Iteration using 3 null vector

	J	$\frac{\partial J}{\partial b_1}$	$\frac{\partial J}{\partial b_2}$	$\frac{\partial J}{\partial b_3}$	b_1	b_2	b_3
Initial	273.45	-0.01	-0.20	-0.01	0	0	0
After	275.62	-6.13E-06	-4.15E-07	9.98E-06	-5.78	-22.65	4.20



Prediction results at s=150, 199, 250 th.



Modification results at s=150, 199, 250 th.

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

- Yuji Shinomura †
- Toru Tamaki †
- Toshiyuki Amano ‡
- Kazufumi Kaneda †

† HIROSHIMA UNIVERSITY

‡ NAIST

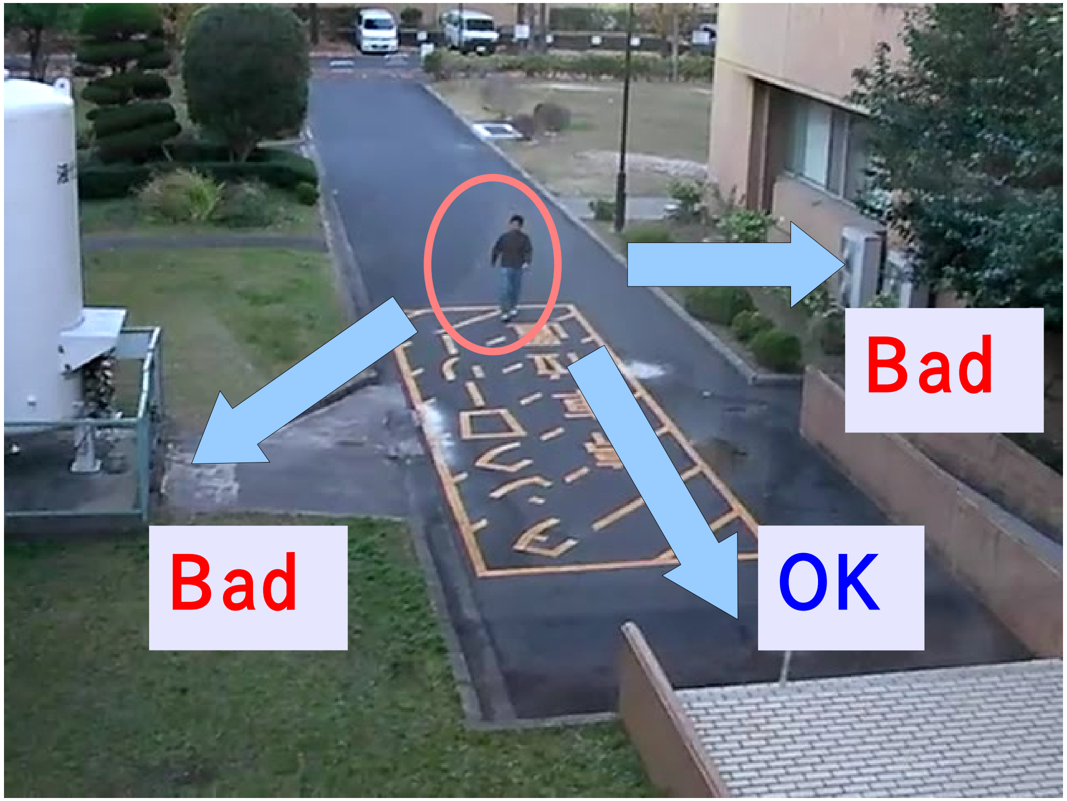
Background

Surveillance camera system

Current : Tracking

Next step …
Judgment of suspicious person

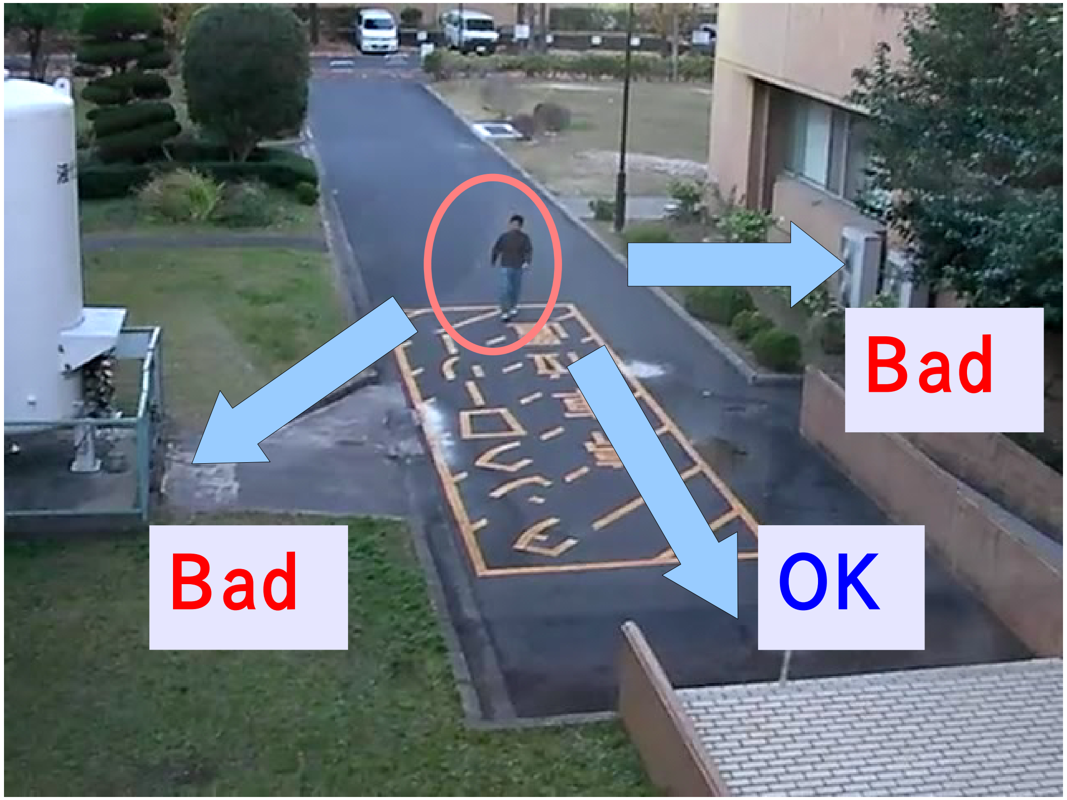
Future : Walking path prediction



Literature review

Path prediction methods

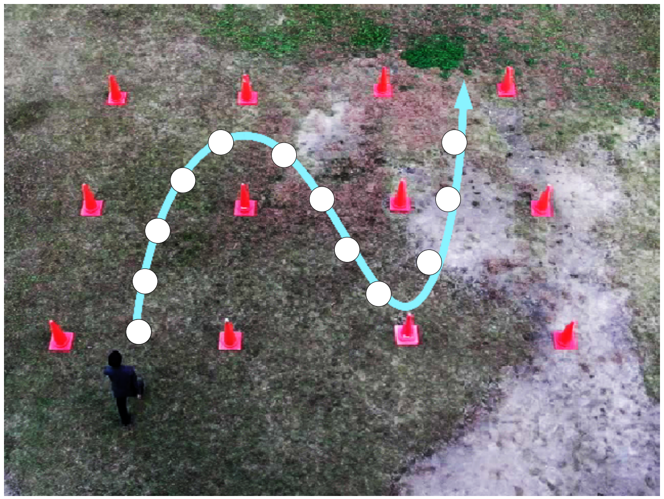
- ✕ – Kalman Filter
- ✕ – Autoregressive(AR) model
- ○ – Eigenspace-based prediction (Yamamoto 2004)



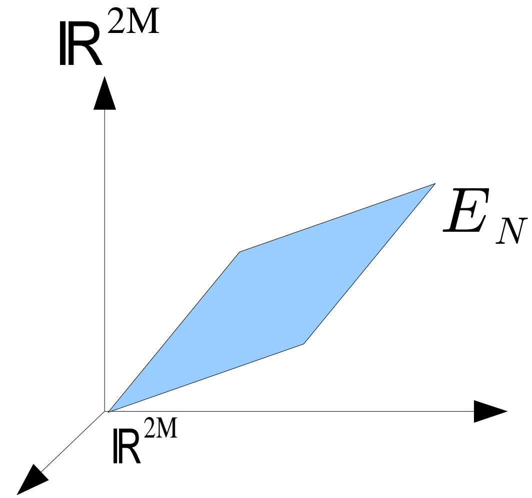
Walking path condition

- Not simple
- Depend on walking environment

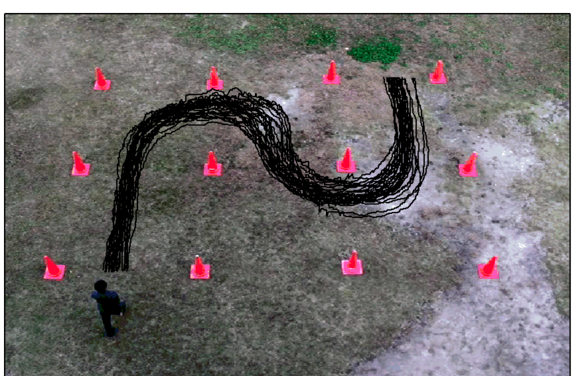
Learning



Walking path : $[\boldsymbol{p}_1^T, \boldsymbol{p}_2^T, \ldots]^T$



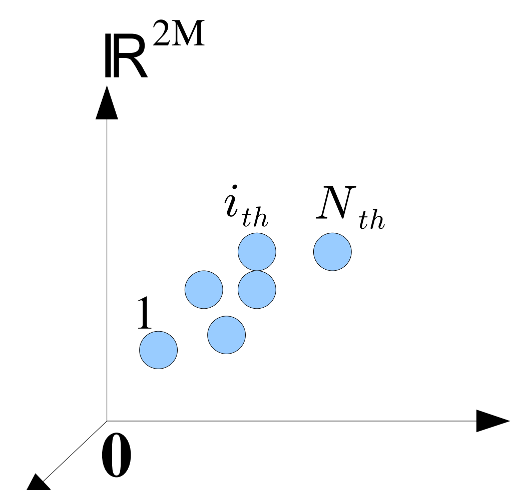
Making Eigenspace



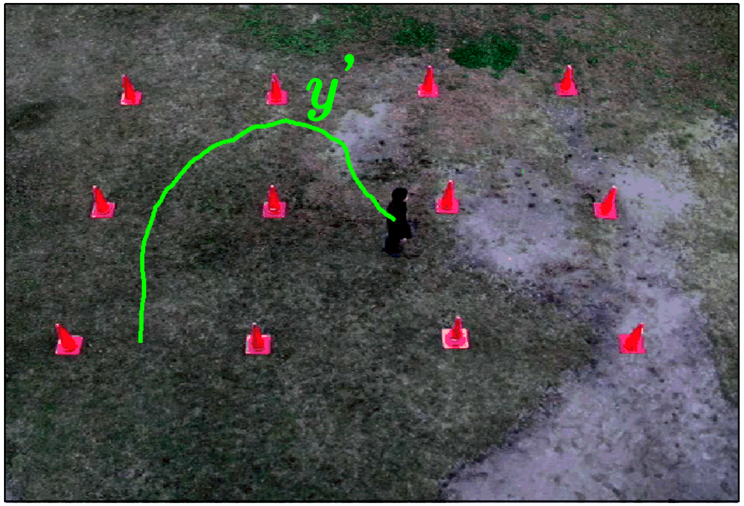
Learning N paths

Normalization

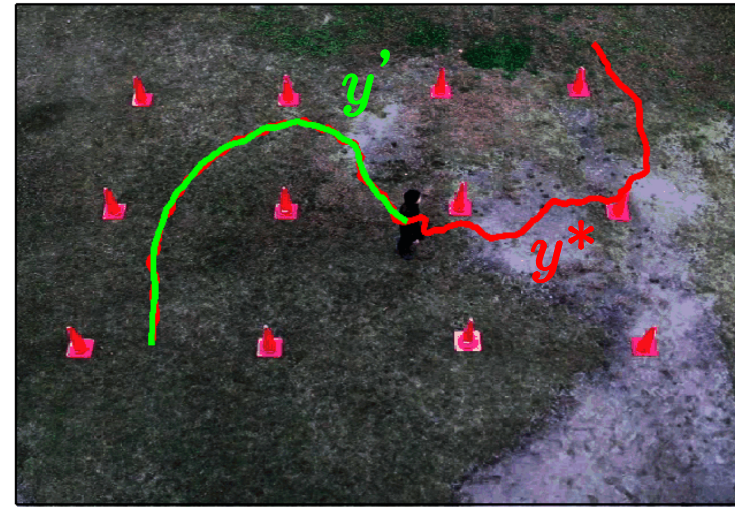
$$\boldsymbol{y} = [\boldsymbol{p}_1^T, \boldsymbol{p}_2^T, \ldots, \boldsymbol{p}_M^T]^T$$



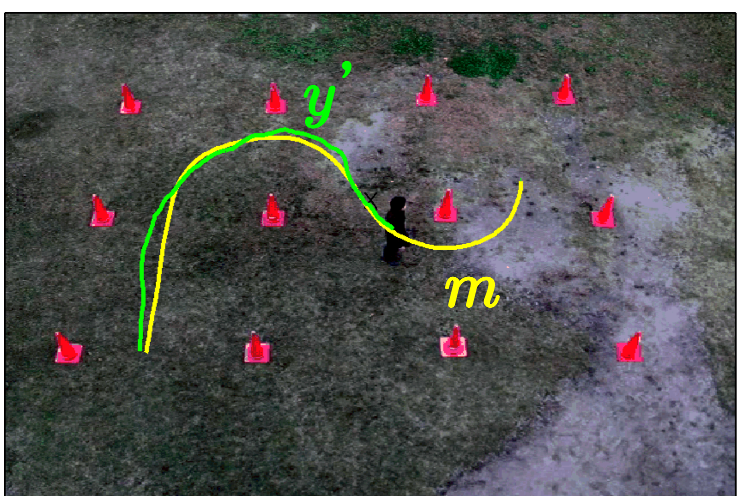
Prediction



$y' = [\boldsymbol{p}_{1,}^T \boldsymbol{p}_{2,}^T \dots, \boldsymbol{p}_s^T]^T \subseteq \mathbb{R}^{2s}$

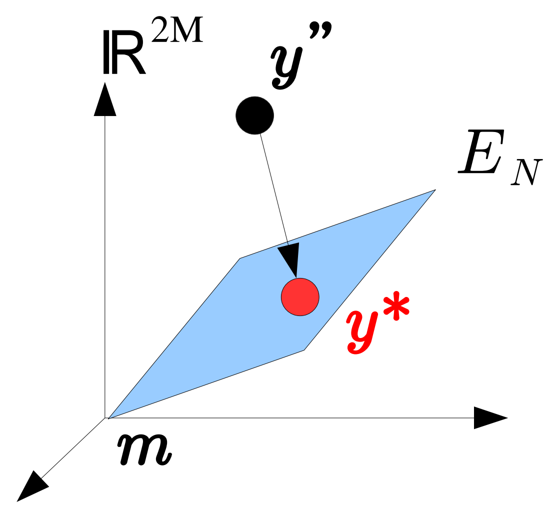


Inverse Projection



$y'' = [\boldsymbol{p}_{1,}^T \boldsymbol{p}_{2,}^T \dots, \boldsymbol{p}_s^T, \boldsymbol{m}_{s+1}, \dots, \boldsymbol{m}_M]^T \subseteq \mathbb{R}^{2M}$

Projection



Problem & Objective

Problem

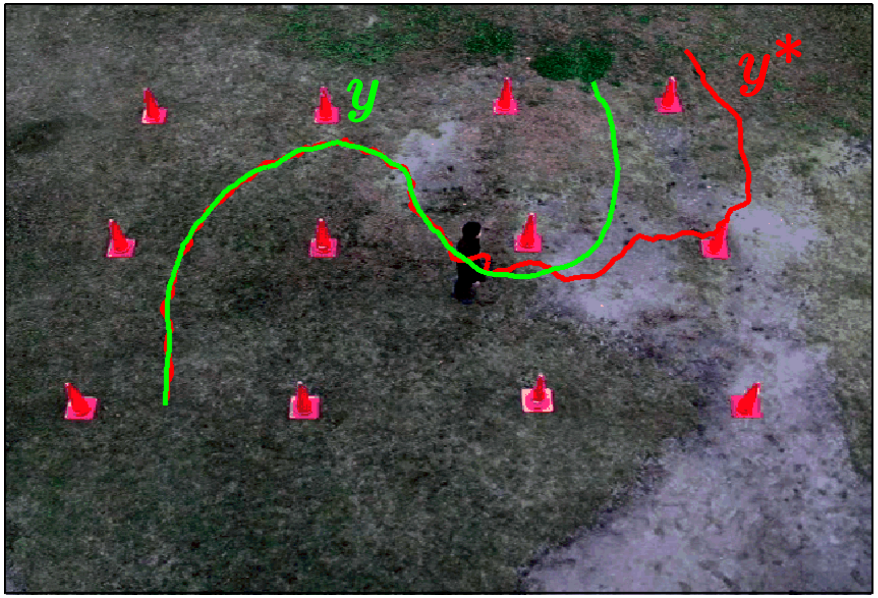
- Prediction is not correspond to actual path

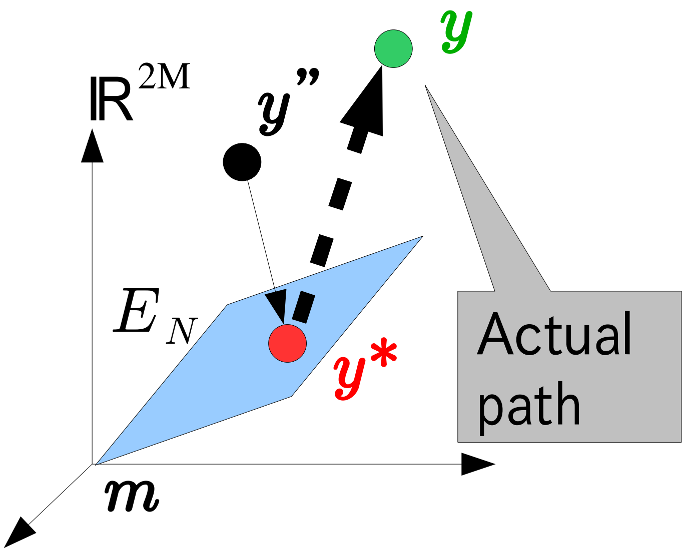
Cause

- Rack of eigenvectors

Objective

- Improvement of prediction result

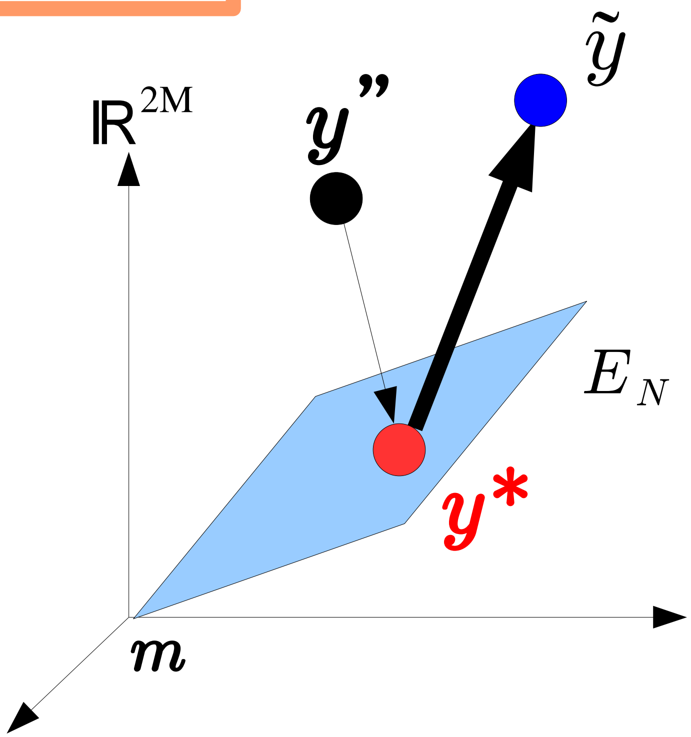




Proposed method

Modifying a Projection using null vector in null space

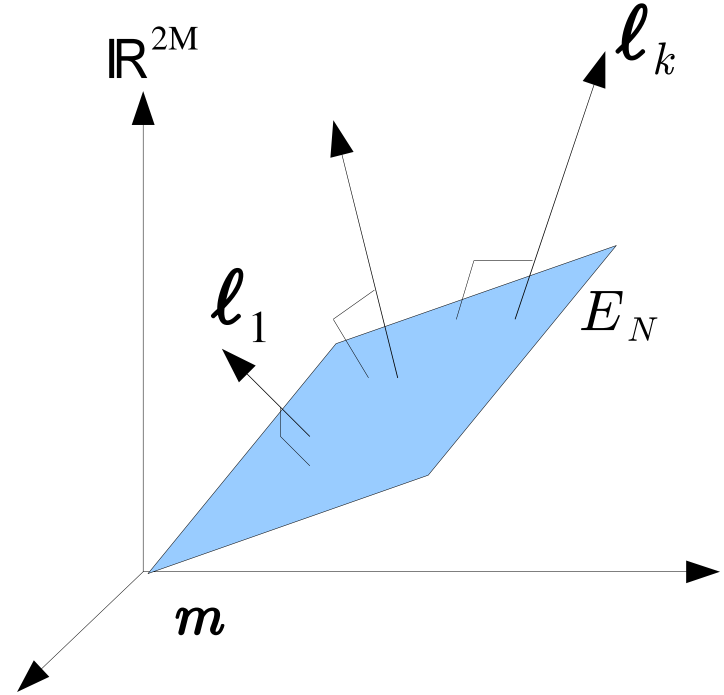
$$\tilde{y} = \underbrace{\sum_{i}^{N} a_i \boldsymbol{e}_i}_{\boldsymbol{y}^*} + \underbrace{\sum_{k}^{s} b_k \boldsymbol{\ell}_k}_{\text{Modified part}}$$



Null vector $\boldsymbol{\ell}_k$

Definition

- A vector Orthogonal of Eigenspace
- Null space $E^{\perp}$ consists of null vectors



Obtainment of null vector

- Using path except for sample path
- Smoothing sample path

Modification using null vector

Assumption

- Walking path is smooth

Cost function

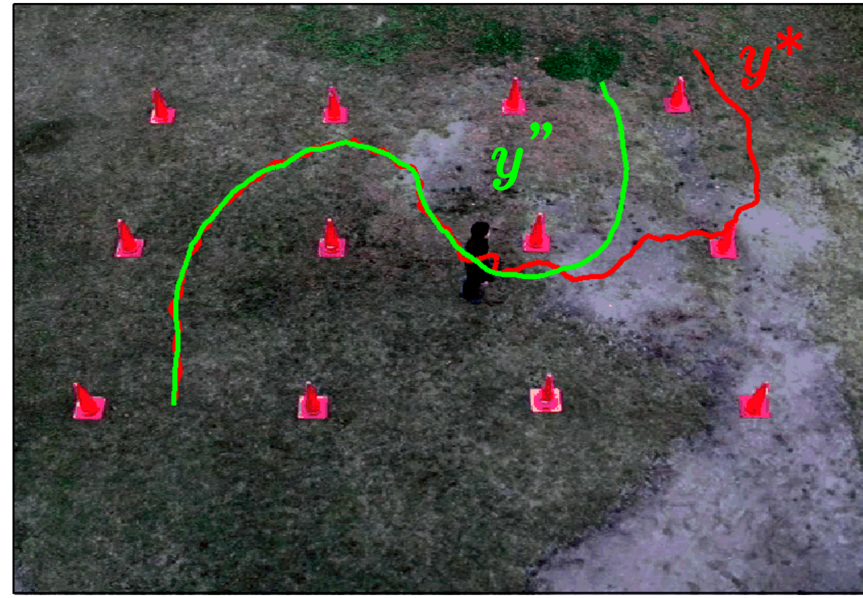
$$maxmize: J = \sum_{t=1}^{M-2} cos^{\alpha}\theta_t$$

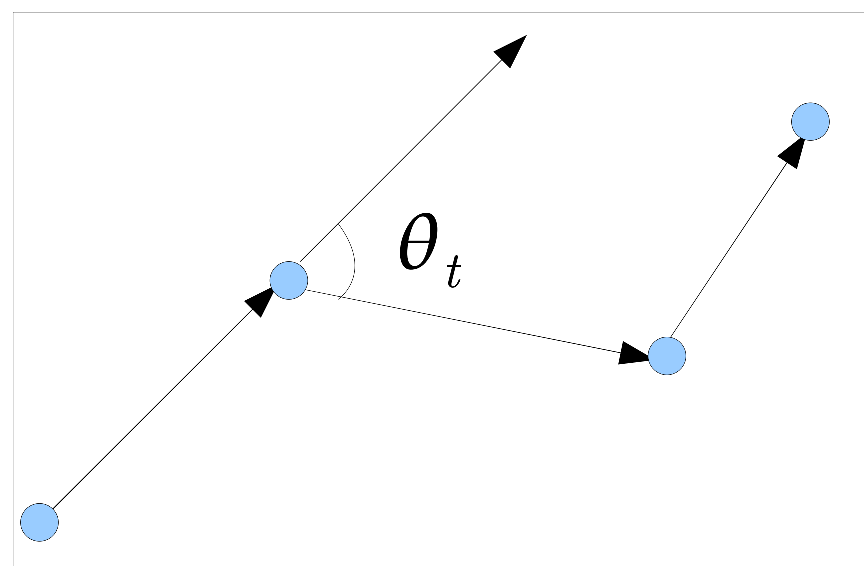
Optimization

- The steepest gradient method

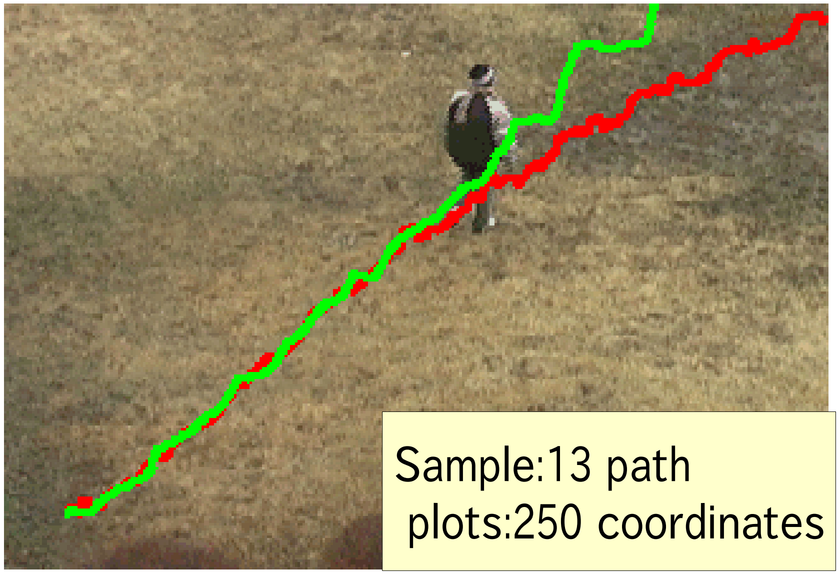
$$b_k \leftarrow b_k + \frac{\partial J}{\partial b_k}$$

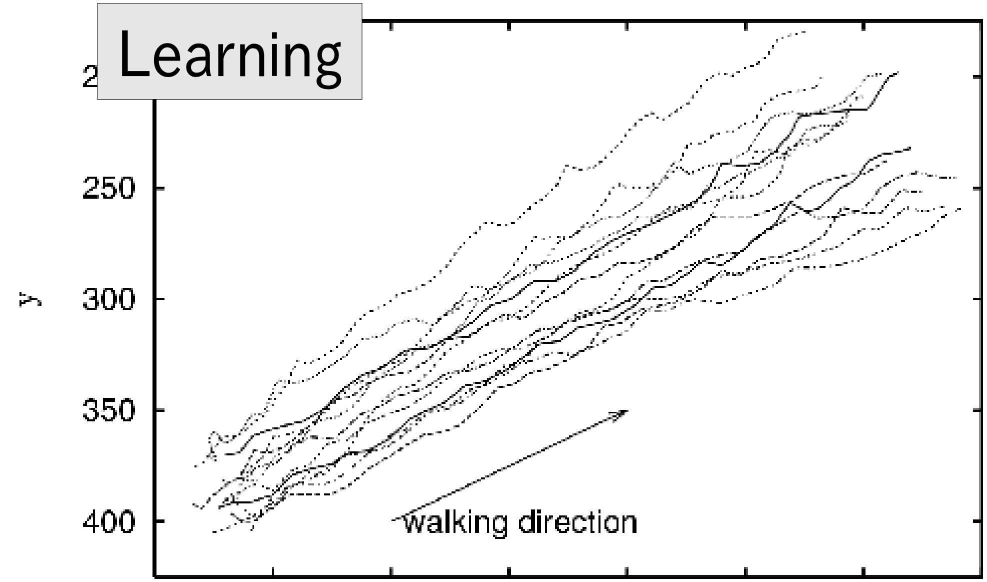
(k: the number of null vector)

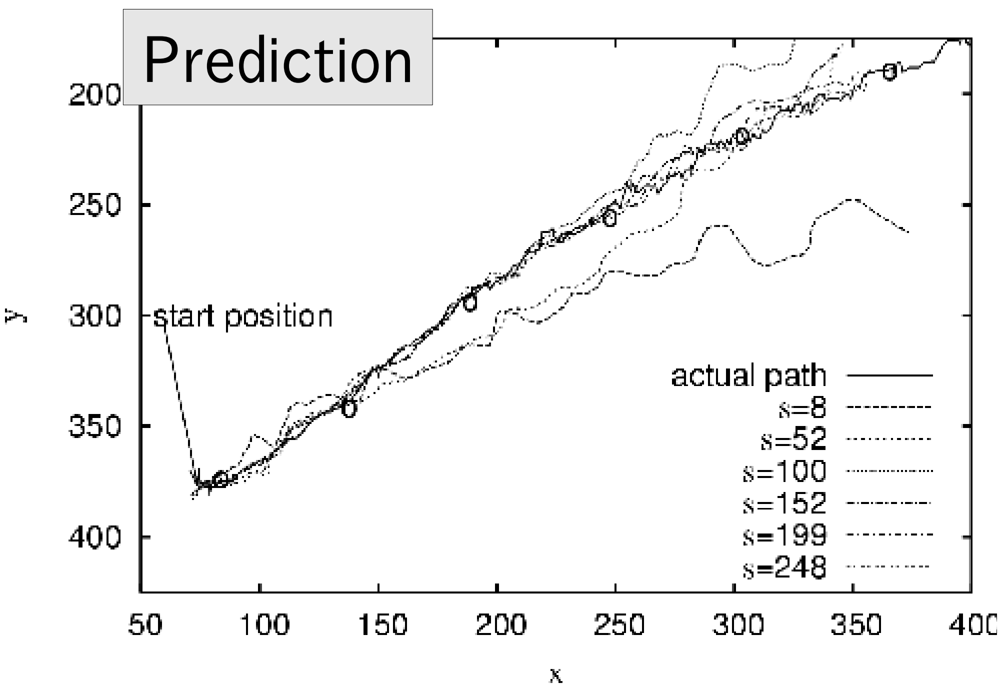


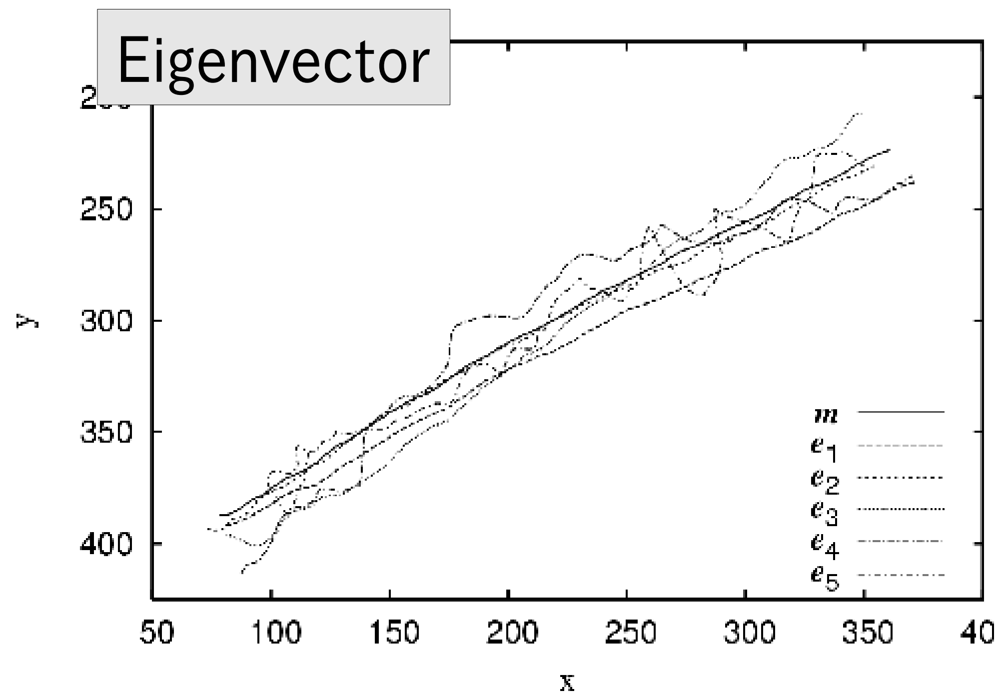


Result :Prediction



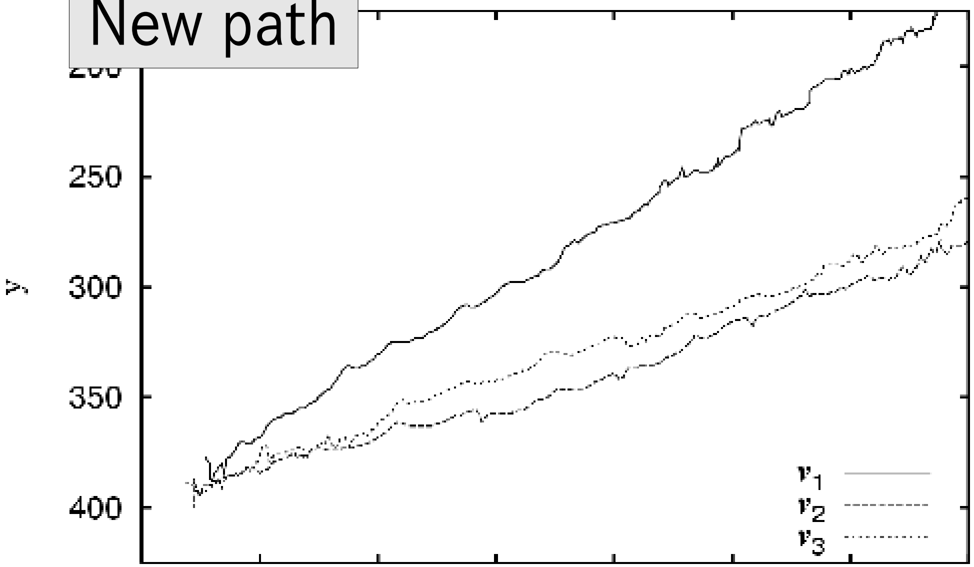




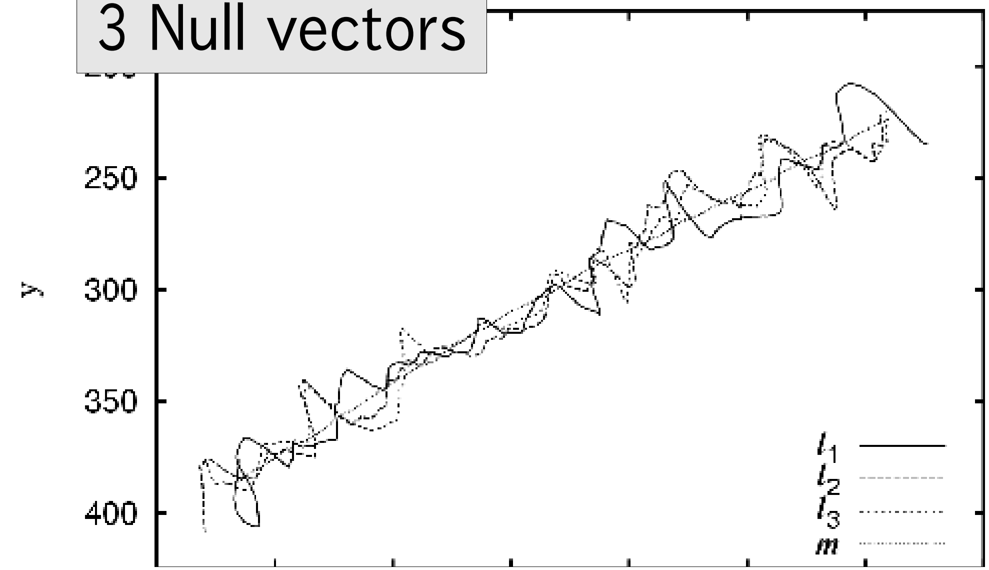


Result : Modification

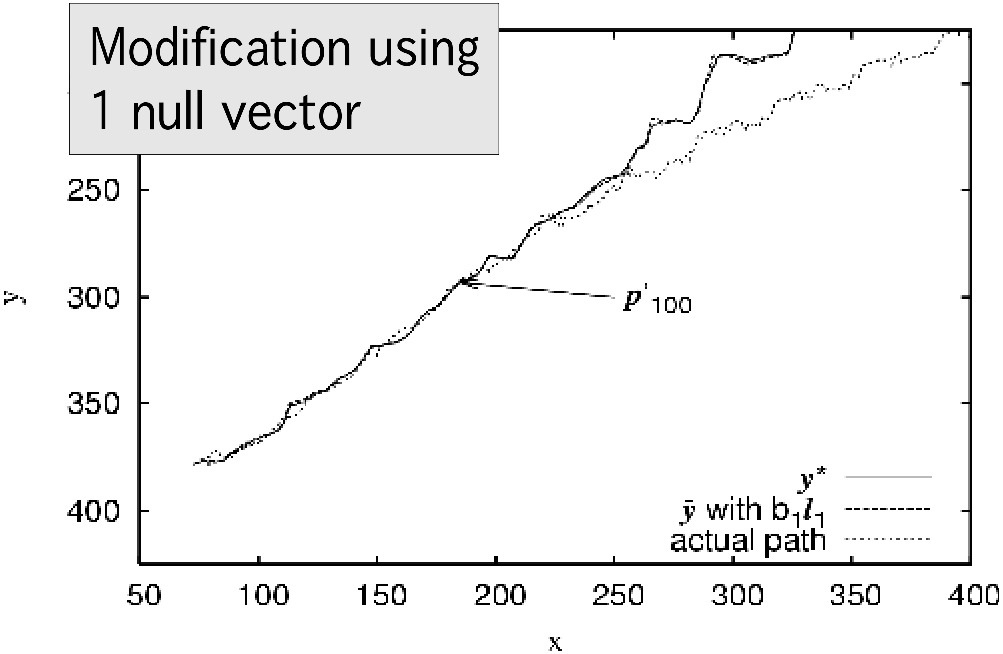
New path



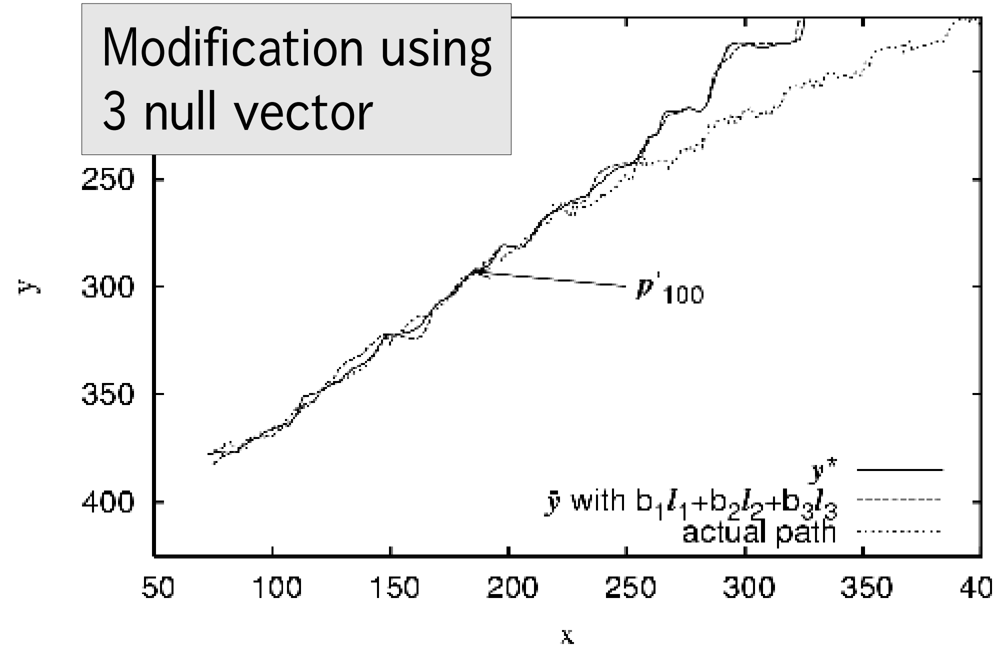
3 Null vectors



Modification using
1 null vector



Modification using
3 null vector



Additional Experiment

Learning

- Sample path : 30
- Downsampling: 50(plots)
- Resampling: 300(plots)

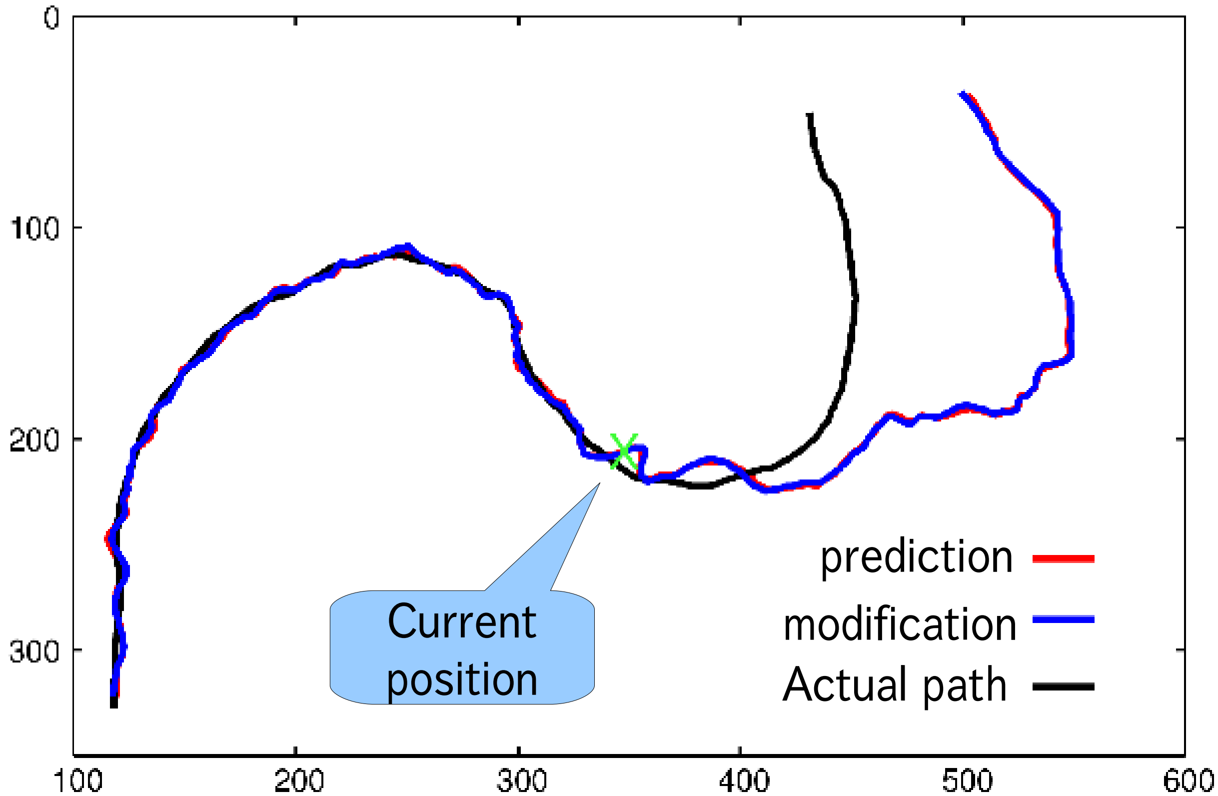
Tracking and Prediction

- Tracking path : 1

Modification

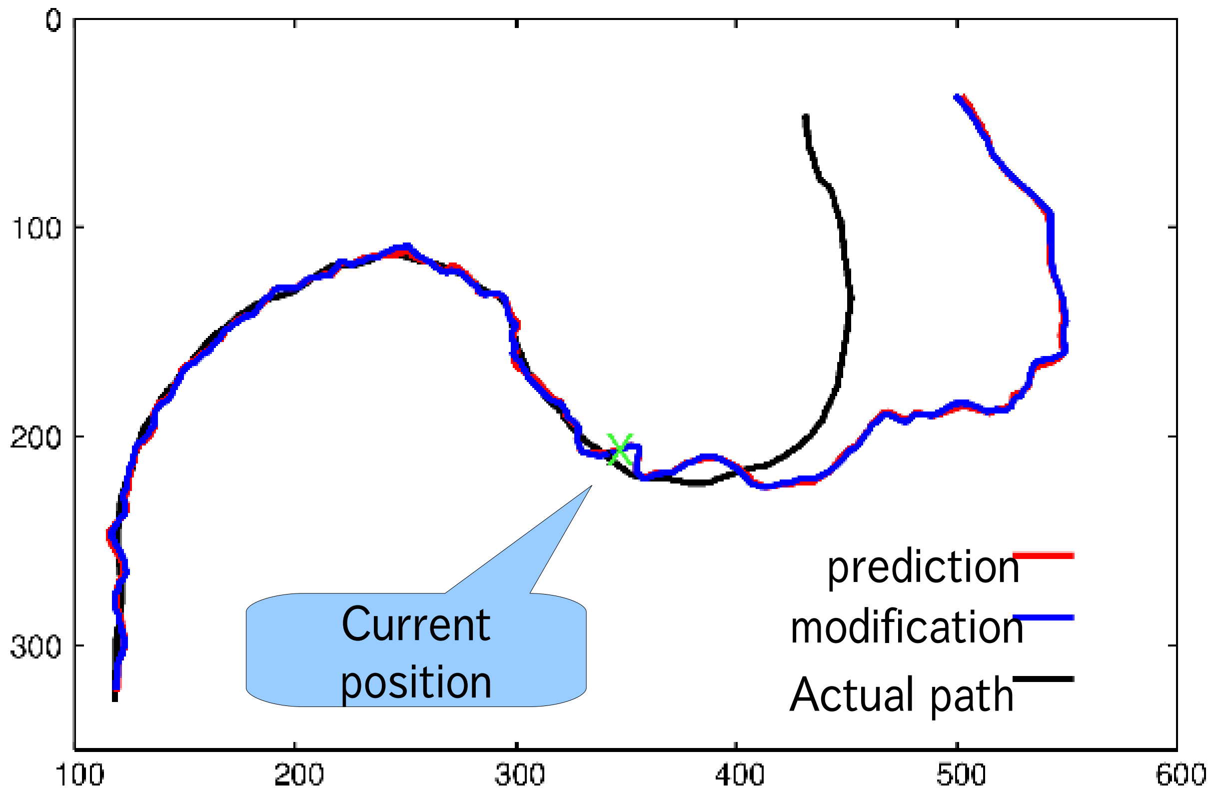
- Null vector : 3 (same course)

Result : Using 1 Null vector



	Initial	After
Cost function: J	273.45	275.52
$\frac{\partial J}{\partial b_k}$	-0.2	-9.95E-06
Coefficient : b_k	0	-22.67

Result :Using 3 null vectors



	J	$\frac{\partial J}{\partial b_1}$	$\frac{\partial J}{\partial b_2}$	$\frac{\partial J}{\partial b_3}$	b_1	b_2	b_3
Initial	273.45	-0.01	-0.20	-0.01	0	0	0
After	275.62	-6.13E-06	-4.15E-07	9.98E-06	-5.78	-22.65	4.20

Conclusions

Summary

- Proposition : Path Modification using Null vector
- Experiments : Not good results

Future works

- Analyzing Effects of type of Null vector
- Making Quantitative Evaluation