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

**Background**

Surveillance camera system

- Current: Tracking
  - Next step: Judgment of suspicious person
  - 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
  \[ \mathbf{p}_n = [\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_s] \subseteq \mathbb{R}^2 \]

Path Normalization

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

Average vector subtraction

- Normalized N sample path are centered by subtracting an average vector \( \mathbf{m} \)

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

3. Average vector subtraction
   - Normalized N sample path are centered by subtracting an average vector \( \mathbf{m} \)

4. Making Eigenspace
   - Singular value decomposition computes eigenvectors \( \mathbf{e}_i \)

Prediction

5. Tracking path \( \mathbf{y}^* \)
   - Person is tracked at \( a \)th frame

6. Compensation
   - Coordinates of 2(\( M - 1 \)) dimension are compensated by average vector \( \mathbf{m} \)
   - Represent on \( \mathbb{R}^{2M} \)

7. Projection onto Eigenspace
   - Linear combination of eigenvectors
   - \( \mathbf{y}^* = E \mathbf{a} \)
   - \( E = \sum_{i=1}^{N} a_i \mathbf{e}_i \)
   - \( E = \text{diag}(1, \ldots, 1, 0, \ldots, 0) \)

8. Inverse projection
   - Add average vector \( \mathbf{m} \) to \( \mathbf{y}^* \)

**Problem & Objective**

**Problem**

- Prediction is not correspond to actual path

**Cause**

- Rack of eigenvectors in (2M-N) Dimension

**Objective**

- Improvement of prediction result
IDEA: Use the orthocomplement of the Eigenspace

\[ \tilde{y} = \sum_{i=1}^{N} a_i \mathbf{e}_i + \sum_{k=1}^{K} b_k \mathbf{e}_k \]

Modified part

What is it needed to use the orthocomplement of the Eigenspace?
- \( \mathbf{e}_k \): Null vector
- \( b_k \): Coefficient of null vector

Null vector \( \mathbf{e}_k \)
- orthogonal vector of Eigenspace.
- Null space \( \perp \): consists of null vectors

How to get Null vector \( \mathbf{e}_k \)

1. Learning new path that is not the same path used in making Eigenspace.
   - Obtaining new walking path
   - Making new path from smoothing learning sample path

2. Subtraction of average vector

3. Gram-Schmidt orthonormalization
   - Making orthogonal vector of Eigenspace and other null vectors
   - Normalization

\[ \mathbf{e}_k = \frac{\mathbf{v}_k}{\| \mathbf{v}_k \|} \]

Cost function
- Consider the degree of smooth of path
- Angle subtended by \( \mathbf{u}_i \) and \( \mathbf{u}_{i+1} \)

\[ \cos \theta \] can be calculated easily as follows:

\[ \cos \theta = \frac{\mathbf{u}_i^T \mathbf{u}_{i+1}}{\| \mathbf{u}_i \| \| \mathbf{u}_{i+1} \|} \]

Finally, the Jacobian of \( J \) comprises \( \mathbf{u}_i \) and \( \mathbf{e}_k \)

How to get coefficient of null vector \( b_k \)

\[ J_{b_k} \] is calculated by

\[ J_{b_k} = \frac{\partial J}{\partial b_k} \]

A stopping condition

\[ \max_k \left| \frac{J}{\| \mathbf{v}_k \|} \right| < 10^{-5} \]

Assumption
- Walking path is smooth

Experimental Results

Case 1:
Learning
- Sample path: 13
- Downsampling: 50(plots)
- Resampling: 250(plots)

Prediction results.

Case 2:
Learning
- Sample path: 30
- Downsampling: 50(plots)
- Resampling: 300(plots)

Modifying 1 null vector and 3 null vectors

Modification result using 3 null vector.

Modification result using 1 null vector.

Prediction result at \( s=150, 199, 250 \)th.

Modification result at \( s=150, 199, 250 \)th.

Coefficient: \( b_1 \), \( b_2 \), \( b_3 \)

Table: Results of Iteration using 1 null vector

<table>
<thead>
<tr>
<th>Initial</th>
<th>After</th>
<th>( J )</th>
<th>( b_1 )</th>
<th>( b_2 )</th>
<th>( b_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial</td>
<td>273.45</td>
<td>0.01</td>
<td>0.01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>After</td>
<td>275.52</td>
<td>0.01</td>
<td>0.01</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table: Results of Iteration using 3 null vector

<table>
<thead>
<tr>
<th>Initial</th>
<th>After</th>
<th>( J )</th>
<th>( b_1 )</th>
<th>( b_2 )</th>
<th>( b_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial</td>
<td>275.62</td>
<td>0.13E-06</td>
<td>0.13E-06</td>
<td>0.13E-06</td>
<td>0.13E-06</td>
</tr>
<tr>
<td>After</td>
<td>275.62</td>
<td>0.13E-06</td>
<td>0.13E-06</td>
<td>0.13E-06</td>
<td>0.13E-06</td>
</tr>
</tbody>
</table>

Coefficient: \( b_1 \), \( b_2 \), \( b_3 \)

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: \([p_1^T, p_2^T, \ldots]^T\)

Learning \(N\) paths

Making Eigenspace

Normalization

\(y = [p_1^T, p_2^T, \ldots, p_M^T]^T\)
Prediction

\[ y' = [p_1^T, p_2^T, \ldots, p_s^T]^T \subseteq \mathbb{R}^{2s} \]

\[ y'' = [p_1^T, p_2^T, \ldots, p_s^T, m_{s+1}^T, \ldots, m_M^T]^T \subseteq \mathbb{R}^{2M} \]
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

\[ \tilde{y} = \sum_{i}^{N} a_i e_i + \sum_{k}^{s} b_k \ell_k \]

\( y^* \)

Modified part

\( \mathbb{R}^{2M} \)

\( E_N \)
Null vector $\ell_k$

**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

\[ \text{maximize } J = \sum_{t=1}^{M-2} \cos^2 \theta_t \]

Optimization

- The steepest gradient method

\[ b_k \leftarrow b_k + \frac{\partial J}{\partial b_k} \]

(\(k\): the number of null vector)
Result: Prediction

Sample: 13 paths, 250 coordinates

Learning

Prediction

Eigenvector
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

Cost function: $J$

<table>
<thead>
<tr>
<th></th>
<th>Initial</th>
<th>After</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J               $</td>
<td>273.45</td>
<td>275.52</td>
</tr>
<tr>
<td>$\frac{\partial J}{\partial b_k}$</td>
<td>-0.2</td>
<td>-9.95E-06</td>
</tr>
<tr>
<td>Coefficient : $b_k$</td>
<td>0</td>
<td>-22.67</td>
</tr>
</tbody>
</table>
Result: Using 3 null vectors

<table>
<thead>
<tr>
<th></th>
<th>$J$</th>
<th>$\frac{\partial J}{\partial b_1}$</th>
<th>$\frac{\partial J}{\partial b_2}$</th>
<th>$\frac{\partial J}{\partial b_3}$</th>
<th>$b_1$</th>
<th>$b_2$</th>
<th>$b_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial</td>
<td>273.45</td>
<td>-0.01</td>
<td>-0.20</td>
<td>-0.01</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>After</td>
<td>275.62</td>
<td>-6.13E-06</td>
<td>-4.15E-07</td>
<td>9.98E-06</td>
<td>-5.78</td>
<td>-22.65</td>
<td>4.20</td>
</tr>
</tbody>
</table>
Conclusions

Summary

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

Future works

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