Robust View-Invariant Representation for Classification and Retrieval of Image and Video Data

BY

XU CHEN
BS, Shanghai Jiaotong University, 2006,
MS, University of Illinois at Chicago, 2009

THESIS

Submitted as partial fulfillment of the requirements for the degree of Electrical and Computer Engineering in the Graduate College of the University of Illinois at Chicago, 2010

Chicago, Illinois
ACKNOWLEDGMENTS

I would like to thank my co-advisors Professor Dan Schonfeld and Professor Ashfaq Khokhar for their guidance, advice and encouragement throughout my graduate study. Their infinite passion and serious attitude in scientific research will benefit me more in my future life.

I also want to thank my doctoral dissertation defense committee, Profs. Shmuel Friedland, Daniela Tuninetti and Milos Zefran for their help and suggestions.

Finally, I would like to thank my parents, for their support at all times.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Main Contributions</td>
<td>2</td>
</tr>
<tr>
<td>1.3 The Organization of the dissertation</td>
<td>3</td>
</tr>
<tr>
<td>2 BACKGROUND AND MOTIVATION</td>
<td>4</td>
</tr>
<tr>
<td>2.1 Related Work</td>
<td>4</td>
</tr>
<tr>
<td>2.2 View Invariant Representation</td>
<td>8</td>
</tr>
<tr>
<td>3 NULL-SPACE INVARIANTS</td>
<td>10</td>
</tr>
<tr>
<td>3.1 Mathematical Formulation</td>
<td>10</td>
</tr>
<tr>
<td>3.2 Perturbation Analysis</td>
<td>13</td>
</tr>
<tr>
<td>3.2.1 2D Perturbation Analysis</td>
<td>13</td>
</tr>
<tr>
<td>3.2.2 Discussion of The Error Ratio in 2D</td>
<td>14</td>
</tr>
<tr>
<td>3.2.3 Discussion of the SNR in 2D</td>
<td>16</td>
</tr>
<tr>
<td>3.2.4 3D Perturbation Analysis</td>
<td>17</td>
</tr>
<tr>
<td>3.2.5 Bounds for the perturbation error</td>
<td>19</td>
</tr>
<tr>
<td>3.3 Sampling Strategy and Convergence Analysis</td>
<td>20</td>
</tr>
<tr>
<td>3.3.1 Uniform sampling</td>
<td>21</td>
</tr>
<tr>
<td>3.3.2 Poisson sampling</td>
<td>22</td>
</tr>
<tr>
<td>3.4 View Invariant Feature Representation for Retrieval and Classification</td>
<td>23</td>
</tr>
<tr>
<td>3.5 View Invariant Motion Trajectories and Images Retrieval and Classification</td>
<td>27</td>
</tr>
<tr>
<td>3.5.1 Dataset</td>
<td>27</td>
</tr>
<tr>
<td>3.5.2 View Invariance for Fixed Cameras From Unknown Views</td>
<td>28</td>
</tr>
<tr>
<td>3.5.3 View Invariance for Arbitrary Moving Cameras</td>
<td>30</td>
</tr>
<tr>
<td>3.5.4 Application in Facial Recognition</td>
<td>42</td>
</tr>
<tr>
<td>3.5.5 Comparison with Traditional Approaches</td>
<td>44</td>
</tr>
<tr>
<td>4 TENSOR NULL-SPACE INVARIANTS</td>
<td>46</td>
</tr>
<tr>
<td>4.1 Introduction</td>
<td>46</td>
</tr>
<tr>
<td>4.2 Tensor Null Space Invariants</td>
<td>47</td>
</tr>
<tr>
<td>4.3 Classification and Retrieval Algorithms</td>
<td>50</td>
</tr>
<tr>
<td>4.4 Simulation Results</td>
<td>52</td>
</tr>
<tr>
<td>5 LOCALIZED NULL-SPACE INVARIANTS</td>
<td>55</td>
</tr>
<tr>
<td>CHAPTER</td>
<td>PAGE</td>
</tr>
<tr>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>5.1</td>
<td>Localized Null Space based Representation</td>
</tr>
<tr>
<td>5.2</td>
<td>Splitting Procedures for LNS</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Deterministic Splitting</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Random Splitting</td>
</tr>
<tr>
<td>5.2.2.1</td>
<td>Optimal Segmentation</td>
</tr>
<tr>
<td>5.2.2.2</td>
<td>Optimal Key Points Selection</td>
</tr>
<tr>
<td>5.3</td>
<td>Merging Procedures for LNS</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Exact Merging</td>
</tr>
<tr>
<td>5.3.1.1</td>
<td>Overlapped Segments</td>
</tr>
<tr>
<td>5.3.1.2</td>
<td>Reconstruction From Key Features Points</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Approximate Merging</td>
</tr>
<tr>
<td>5.4</td>
<td>Perturbation Analysis</td>
</tr>
<tr>
<td>5.5</td>
<td>Performance Evaluation</td>
</tr>
<tr>
<td>5.5.1</td>
<td>Simulation Results for Splitting</td>
</tr>
<tr>
<td>5.5.2</td>
<td>Simulation Results for Merging</td>
</tr>
<tr>
<td>6</td>
<td>NON-LINEAR KERNEL SPACE INVARIANTS</td>
</tr>
<tr>
<td>6.0.3</td>
<td>Basis Decomposition</td>
</tr>
<tr>
<td>7</td>
<td>MULTILINEAR INVARIANTS</td>
</tr>
<tr>
<td>7.1</td>
<td>Introduction</td>
</tr>
<tr>
<td>7.2</td>
<td>Invariants Theory</td>
</tr>
<tr>
<td>7.2.1</td>
<td>T-Invariants</td>
</tr>
<tr>
<td>7.2.2</td>
<td>Linear Invariants</td>
</tr>
<tr>
<td>7.2.3</td>
<td>Bilinear Invariants</td>
</tr>
<tr>
<td>7.3</td>
<td>Tensor Multilinear Invariants</td>
</tr>
<tr>
<td>7.4</td>
<td>Reduced-Dimension Multilinear Invariants</td>
</tr>
<tr>
<td>8</td>
<td>CONCLUSION AND FUTURE WORK</td>
</tr>
<tr>
<td>9</td>
<td>REFERENCES</td>
</tr>
<tr>
<td>10</td>
<td>APPENDICES</td>
</tr>
<tr>
<td>10.1</td>
<td>The proof for the convergence of the mean of the error ratio with Poisson sampling</td>
</tr>
<tr>
<td>10.2</td>
<td>The proof for the convergence of SNR with uniform sampling</td>
</tr>
<tr>
<td>10.3</td>
<td>The proof for the convergence of the variance of the error ratio with Poisson sampling</td>
</tr>
<tr>
<td>11</td>
<td>VITA</td>
</tr>
<tr>
<td>TABLE</td>
<td>PAGE</td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
</tr>
<tr>
<td>I</td>
<td>34</td>
</tr>
</tbody>
</table>

**TABLE I**
COMPARISON OF CLASSIFICATION ACCURACY FOR GLOBAL TRAJECTORIES AND SEGMENTED TRAJECTORIES FOR 20 CLASSES ON ASL AND CAVIAR DATASET
<table>
<thead>
<tr>
<th>FIGURE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>30</td>
</tr>
<tr>
<td>3</td>
<td>31</td>
</tr>
<tr>
<td>4</td>
<td>31</td>
</tr>
<tr>
<td>5</td>
<td>33</td>
</tr>
<tr>
<td>6</td>
<td>35</td>
</tr>
<tr>
<td>7</td>
<td>36</td>
</tr>
<tr>
<td>FIGURE</td>
<td>PAGE</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>8</td>
<td>Three sample trajectories from ASL dataset. For each trajectory, another two versions of noisy trajectories with arbitrary moving cameras are also shown. Original trajectories are drawn in blue with the variance of noise 0.02. .................. 37</td>
</tr>
<tr>
<td>9</td>
<td>Visual illustration for an example of three versions of the same noisy trajectory from the class &quot;all&quot; in ASL dataset with and without Poisson sampling ($\lambda = 0.8$) respectively. Typically, (a),(b),(c) are the representations without Poisson samplings; (d),(e),(f) are the representations with Poisson samplings. ................................ 38</td>
</tr>
<tr>
<td>10</td>
<td>Comparison of null space representation for three versions of the same noisy trajectory from the class &quot;all&quot; in ASL dataset with and without Poisson sampling (the first 16 nonzero elements). .................. 39</td>
</tr>
<tr>
<td>11</td>
<td>The block diagram of the proposed classification and retrieval system . 39</td>
</tr>
<tr>
<td>12</td>
<td>24 different face poses used for each face from the UMIST face database 40</td>
</tr>
<tr>
<td>13</td>
<td>(a) Feature points extraction for the 1st and 6th frames based on SIFT in the UMIST face database, (b) Accuracy of classification with increasing number of classes, (c) Accuracy for classification problems with the increase in the number of images within a class, (d) Precision-Recall metric by using PCNSA on NSI for original and noisy images. .......... 41</td>
</tr>
<tr>
<td>14</td>
<td>(a) The comparison of PR curve for our approach and curvature based approach for motion trajectory retrieval for perfect trajectories, (b) The comparison of PR curve for our approach and CSS-based approach in retrieval for noisy trajectories. ................................. 45</td>
</tr>
<tr>
<td>15</td>
<td>Flattening a 3rd order motion event tensor for multiple trajectories representation. ................................ 53</td>
</tr>
<tr>
<td>16</td>
<td>Accuracy for tensor (multiple motion trajectories) classification (a) with an increasing number of classes and (b) with an increasing number of tensors within a class. ................................. 54</td>
</tr>
<tr>
<td>17</td>
<td>Visual illustrations of the original trajectory and part of the rotated motion trajectory with identical localized null space representation from CAVIAR data set. ................................. 63</td>
</tr>
<tr>
<td>FIGURE</td>
<td>PAGE</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>18</td>
<td>64</td>
</tr>
<tr>
<td>19</td>
<td>75</td>
</tr>
<tr>
<td>20</td>
<td>76</td>
</tr>
<tr>
<td>21</td>
<td>77</td>
</tr>
<tr>
<td>22</td>
<td>77</td>
</tr>
<tr>
<td>23</td>
<td>78</td>
</tr>
<tr>
<td>24</td>
<td>78</td>
</tr>
<tr>
<td>25</td>
<td>86</td>
</tr>
<tr>
<td>26</td>
<td>87</td>
</tr>
</tbody>
</table>

The illustrations of the structures of the traditional Null Space and the purposed Localized Null Space.

Top three retrieval results for a partial query trajectory containing only 75% samples with the missing portion from CAVIAR dataset.

Top five retrievals corresponding to the partial query for motion trajectory from ASL dataset which losts 10% samples.

Top three retrieval results for merging three sub-trajectories as the query containing only 60% samples.

The comparison of precision and recall curves in retrieval with curvature scale space (CSS) and centroid distance function (CDF) based approaches.

The comparison of average precisions for retrieval of different types of motion events in CAVIAR dataset with 20% missing data.

The comparison of cpu time for merging process with different number of segments using ASL dataset.

Unequal multiple trajectory representation: A video frame and its rotated version containing three trajectories with significantly different length, where trajectories A, B, and C have 45, 12, and 26 samples, respectively.

Multi-camera multiple trajectory representation: Two videos for the same motion event captured from different moving cameras containing 18 trajectories in each video sequence and the maximum length of the trajectories is 15.
<table>
<thead>
<tr>
<th>FIGURE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMS</td>
<td>American Mathematical Society</td>
</tr>
<tr>
<td>CTAN</td>
<td>Comprehensive \TeX\ Archive Network</td>
</tr>
<tr>
<td>TUG</td>
<td>\TeX\ Users Group</td>
</tr>
<tr>
<td>UIC</td>
<td>University of Illinois at Chicago</td>
</tr>
<tr>
<td>UICTHESI</td>
<td>Thesis formatting system for use at UIC.</td>
</tr>
</tbody>
</table>
SUMMARY

Representation, storage, retrieval, and classification of motion events consisting of multiple objects captured by multiple cameras is an extremely challenging task. Existing modelling and classification techniques are inadequate and highly unscalable to handle this challenge. In this dissertation we investigate the design of a novel robust view-invariant retrieval and classification system for video and motion events based on null-space representation of these events. We first derive the Null Space Invariant (NSI) matrix representation for motion trajectories. The proposed representation based on the NSI operator is invariant to affine transformation. We subsequently rely on principal component null space analysis (PCNSA) for indexing and classification of the NSI operator, and demonstrate the efficiency of the null space representation in recognition and retrieval of motion events.

We also analyze the robustness of the null space representation to noise using perturbation theory. We derive the error ration and SNR as measures of the sensitivity of the null operators. Given the perturbation, we prove that uniform sampling and Poisson sampling lead to convergence of the SNR and error ratio, respectively. The simulation results demonstrate the effectiveness and robustness of our system in motion event indexing, retrieval and classification in the presence of affine transformations due to camera motion.

We also develop a general framework for tensor-based null-space affine invariants, namely, tensor null space invariants (TNSI) to represent motion events consisting of multiple interacting objects. We have proposed to rely on a linear classifier for high-order data classification and
summaries (Continued)

retrieval. We prove that TNSI is invariant to multi-dimensional affine transformations due to camera motion for multiple motion trajectories in consecutive motion events. Based on TNSI, we propose an efficient classification and retrieval system. Through simulations using real-life dataset and demonstrate the superiority of the proposed system compared to existing state of the art using simulation results.

Next, we consider splitting and merging of null-space view-invariant representations in dynamic video databases. The framework provides a novel method for view-invariant classification and retrieval with partial queries and dynamic data updating. The proposed approach is the only method known to us that can handle dynamic update and view invariance within a single framework. We present an exact solution for merging multiple null spaces into an integrated null space representation without requiring access to the raw data, thus allowing dynamic updating of the database. We also present a novel splitting algorithm to extract null space representations of subspaces from an existing null space representation, thus allowing classification and retrieval using partial information. We demonstrate the effectiveness of the proposed techniques for motion event classification and retrieval applications by posing different affine transformed partial queries. The comparison of performance with traditional approaches demonstrates the promising results of the proposed approach.

Finally, we develop the Non-linear Kernel Space Invariants (NKSI) for non-linear transformation of the raw data and Multilinear Invariants for view invariant retrieval of raw data with unequal length of different dimensions. We also extend the concept of Multilinear Invariants to
SUMMARY (Continued)

Tensor Multilinear Invariants for high dimensional data. We provide the simulation results to demonstrate the effectiveness of our approach.
CHAPTER 1

INTRODUCTION

1.1 Introduction

Video based event recognition is an extremely challenging task due to all kinds of within-event variations, such as unconstrained motions, cluttered backgrounds, object occlusions, environmental illuminations and geometric deformations of objects. Object motion-based analysis and recognition has gained significant interest in scientific circles lately. This trend can be attributed mainly to unprecedented advances in hardware and software technologies that allow spatiotemporal data of objects to be easily derived from video and non-video sources. Also, novel applications employing analysis of motion trajectory are emerging due to enhanced interest in homeland security as well as due to prevalence of multimedia gadgets in commercial and scientific endeavors. Examples of the motion trajectory include tracking results from video trackers, sign language data measurements gathered from wired glove interfaces fitted with sensors, Global Positioning System (GPS) coordinates of satellite phones, etc. An important application area in this domain is automatic video surveillance which is used, for example, in real-time observation of people and vehicles, in a busy environment, leading to a description of actions and mutual interactions. The research challenge here is to quickly learn the permitted activities and set an alarm at any illegal or abnormal activity being performed. We emphasize that in light of psychological studies reported in the literature, it is clear that object motion
plays a key role in the domain of activity analysis in general and in video surveillance in particular.

We observe that developing high-accuracy activity classification and recognition algorithms in multiple view situations is still an extremely challenging task. The object trajectory is typically modeled as a sequence of consecutive locations of the object on a coordinate system resulting in a vector in 2-D or 3-D Euclidean space. An object trajectory captured from different viewpoints leads to entirely different representations. Multiple representations of an object motion captured from different view points are related by projective transformation exactly, and by generic affine transformation approximately. The set of affine transformations includes scaling, rotation, translation and shear. To satisfy the view independent requirement, the trajectory data has to be represented in an affine invariant feature space.

1.2 Main Contributions

The main contributions in this dissertation include:

1. Develop a novel framework for null space view invariant representation of image and video classification and retrieval.

2. Analyze the robustness of the null space invariants relying on perturbation analysis.

3. Prove the convergence of the error ratio and SNR of the null space invariants with perturbation with optimal sampling: uniform sampling and Poisson sampling.

4. Propose a novel framework based on tensor null space for multi-dimensional data classification and retrieval.
5. Propose a novel framework based on localized null space for view invariant representation in the dynamical image and video database with partial queries.

6. Propose a novel framework based on non-linear kernel space invariants which provide an invariant representation for non-linear transformations.

7. Propose a novel framework based multilinear invariants which provide an invariant representation for the data with unequal length.

1.3 The Organization of the dissertation

The rest of the dissertation is organized as follows: in Chapter 2, we provide a brief review of the background in the area of the motion trajectory analysis and the motivation of our work. In Chapter 3, we propose the novel framework based on null space invariants for image and video retrieval. We analyze the robustness of null space invariants with perturbation analysis. We further prove the convergence of the error ratio and SNR for perturbed null space with uniform sampling and Poisson sampling. In Chapter 4, the novel framework of tensor null space for classification and retrieval of multi-dimensional data is proposed. In Chapter 5, the novel frame of localized null space for dynamical updating and downdating of null space is investigated. In Chapter 6 and 7, the novel framework of non-linear kernel space invariants and multilinear invariants are discussed. We give the conclusion and discuss about the future work in Chapter 8.
CHAPTER 2

BACKGROUND AND MOTIVATION

2.1 Related Work

Within the last several years, object motion trajectory-based recognition has gained significant interest in diverse application areas including sign language gesture recognition, Global Positioning System (GPS), Car Navigation System (CNS), animal mobility experiments, sports video trajectory analysis and automatic video surveillance (16) (34) (35) (36) (37). Psychological studies show that human beings can easily discriminate and recognize an object’s motion pattern even from large viewing distances or poor visibility conditions where other features of the object vanish. Object motion is an important feature for the representation and discrimination of an object and its activities from others in video applications. Earlier approaches in motion-based methods focused on object tracking from raw and compressed domain videos. Indexing and searching based on object motion as the dominant cue has attracted a lot of research activity in the past few years. The previous work on trajectory indexing and retrieval segments the trajectories based on dominant sign changes in curvature data. They represent the sub-trajectories using PCA coefficients. The work has addressed the view-invariant representation of trajectories for scenarios where similar trajectories are captured from different view points. Recently, semantics-based processing of trajectory data to extract high-level information has gained significant interest. The development of accurate activity classification and recognition
algorithms in multiple view situations is still an extremely challenging task. Object trajectories captured from different view-points lead to completely different representations, which can be modeled by affine transformation approximately. To get a view independent representation, the trajectory data is represented in an affine invariant feature space.

We provide a survey of the related work from recent literature in the areas of motion feature computation for trajectory representation, principal component analysis and applications of trajectory-based indexing and retrieval. Studies into human psychology have shown the extraordinary ability of human beings to recognize object motion even from minimal information system such as moving light displays (MLDs) (11). Given its importance, MPEG-7 adopted the notion of motion activity and motion trajectory in a collection of motion descriptors that capture the different aspects of motion in videos with a broad range of precision (15). The standard defines concise descriptors including motion activity and motion trajectory that are easy to extract and match without providing the details of the indexing and retrieval process.

Object motion has been an important feature for object representation and activity modeling in video applications. An object trajectory-based system for video indexing is proposed, in which the normalized - and -projections of trajectory are separately processed by wavelet transform using Haar wavelets. Chen et al (13) segment each trajectory into subtrajectories using fine-scale wavelet coefficients at high levels of decomposition. A feature vector is then extracted from each subtrajectory and Euclidean distances between each subtrajectory in the query trajectory and all the indexed subtrajectories are computed to generate a list of similar trajectories in the database. The longest common subsequence (LCSS) approach (9) is
used for grouping similar motion trajectories in an agglomerative clustering algorithm. The
LCSS is defined recursively as increasing distance between two sequences based on their - and
-projections. The similarity is then computed from the two sequences as the least distance
under a set of translations. Shim (10) proposed a modification of the DTW algorithm using
a k-warping distance algorithm by permitting up to k replications for an arbitrary motion of a
query trajectory to measure the similarity between two trajectories. Their approach is tested
on a content-based soccer video retrieval (CSVR) system in which the trajectory of the soccer
ball is extracted by manually tracking the ball in a ground field using linear segments. Ob-
ject trajectory data can be viewed as a time series when - and -projections are combined for
representation. There has been tremendous amount of activity in time series representation
and retrieval in recent years. Lin et al (7)(8) have presented a symbolic representation of a
time-series approach (SAX) using piecewise aggregate approximation (PAA). Although quite
close to our string matching-based system, there are two major problems with it for trajectory
data. The PAA uses fixed box-bases to represent continuous noisy data which might not be
very suitable for most time series. Also, the notion of fixed breakpoints used in their approach
to map time series segments to symbols is questionable as it is perceptually not quite appealing
for trajectory data. Fink et al. (6) represent time series by maxima, minima and major inclines.
They identify all major inclines of all series in the dataset and index them using a range tree.
Their piecewise linear approximation is similar to polynomial approximation which was shown
to perform worse than our PCA-based representation. Recently, significant research effort has
gone into devising new space- and time- efficient index structures. An important application
area of trajectory-based indexing is human activity modeling. Yacoob et al. (14) have presented a framework for modeling and recognition of human motions based on principal components. Each activity is represented by eight motion parameters recovered from five body parts of the human walking scenario. The high-dimensional trajectory using all the eight parameters of object motion is reduced using PCA. In (38), the issue of recognizing a set of plays from American football videos is considered. Using a set of classes each representing a particular game plan and computation of perceptual features from trajectories, the propagation of uncertainty paradigm is implemented using automatically generated Bayesian network. On similar lines, Nevatia et al. (5) have addressed the issue of activity recognition in single or multiple actor situations which exhibit some specific patterns of whole body motion.

A new invariant algorithm for Structure From Motion (SFM) problem is proposed in (17). It uses invariant property of group action on a vector space to eliminate the camera pose parameters in the calculations. That enables robust solutions for SFM. Although the invariant algorithm is promising to use in recognition and classification problems, the requirement for solving the invariant equations by Levenberg-Marquardt algorithm introduces high computational complexity. In (16), two different affine invariant representations for motion trajectories namely: Curvature Scale Space (CSS) and Centroid Distance Functions (CDF) have been used for trajectory recognition and classification. However, they are view invariant only when the camera motions are small. Other work includes the use of kernel methods with multilevel temporal alignment (4). However, these work is not view invariant.

In this dissertation, we introduce a simple but highly efficient view invariant representation
based on Null Space Invariant (NSI) matrix. As far as we know, this is the first use of Null space in motion-based classification/retrieval applications.

Indexing and classification of the NSI operator is obtained by extracting features of the null space representation using Principal Components Null Space Analysis (PCNSA), which provides an efficient analysis tool when different classes may have different non-white noise covariance matrices (26). Dimensionality reduction for indexing of the NSI is achieved by first performing Principal Components Analysis (PCA) as part of PCNSA. Classification is performed in PCNSA by determining the \( i \)th class \( M_i \)-dimensional subspace by choosing the \( M_i \) eigenvectors that give the smallest intra-class variance. The \( M_i \)-dimensional space is referred to as the Approximate Null Space (ANS). A query is classified into the class if its distance to the class mean in ANS space is lowest among all the other classes.

2.2 View Invariant Representation

In (18), the mathematical form of the representation of the null space invariants has been derived. In this dissertation, we will rely on the theoretical formulation of the null space invariants to demonstrate its enormous potential in computer vision and related fields. Specifically, we will demonstrate the invariance of the null space representation of motion trajectories of moving objects to camera orientation and movement. We will subsequently investigate the robustness of the proposed approach to view invariance based on null space representation. We will evaluate the sensitivity of trajectory analysis to noisy data by using tools from perturbation theory. We will determine the performance of the null space representation in terms of both the error ratio and SNR. We will finally derive the optimal sampling strategy that minimizes the
sensitivity to noise perturbation and thus ensures the robustness of null space representation. Furthermore, we will demonstrate that the optimal sampling strategy required to maximize the SNR is provided by uniform sampling. We therefore observe that the null space representation is particularly suitable for the invariant representation of computer vision systems since they traditionally rely on uniform sampling for data acquisition (e.g., uniform sampling of the temporal dimension is used to capture video signals).
CHAPTER 3

NULL-SPACE INVARIANTS

3.1 Mathematical Formulation

A fundamental set of 2-D affine invariants for an ordered set of n points in $\mathbb{R}^2$ (not all colinear) is expressed as an n-3 dimensional subspace, $H^{n-3}$, of $\mathbb{R}^{n-1}$, which yields a point in the 2n-6 dimensional Grassmannian $\text{Gr}_{\mathbb{R}}(n-3, n-1)$, a manifold of dimension 2n-6. Null space invariant (NSI) of a trajectory matrix (each row in the matrix corresponds to the positions of a single object over time) is introduced as a new and powerful affine invariant space to be used for trajectory representation. This invariant, which is a linear subspace of a particular vector space of a particular vector space, is the most natural invariant and is definitely more general and more robust than the familiar numerical invariants. It does not need any assumptions and after invariant calculations it conserves all the information of the original raw data. Let $Q_i = (x_i, y_i)$ be a 2-D point for $i=0,1,...,N-1$ N ordered non-linear points in $\mathbb{R}^2$. Consider the $3 \times N$ matrix $M$ as:

$$M = \begin{pmatrix}
  x_0 & x_1 & \ldots & x_{N-1} \\
  y_0 & y_1 & \ldots & y_{N-1} \\
  1 & 1 & \ldots & 1
\end{pmatrix}.$$  \hspace{1cm} (3.1)
As described in (18), $H^{N-3}$ is spanned by the vector $v_i = (q_0^i, q_1^i, ..., q_{N-1}^i)^T$, $i=3,4,...,N-1$, where

$$q_0^i = -\det \begin{pmatrix} x_1 & x_2 & x_i \\ y_1 & y_2 & y_i \\ 1 & 1 & 1 \end{pmatrix}$$

$$q_1^i = \det \begin{pmatrix} x_0 & x_2 & x_i \\ y_0 & y_2 & y_i \\ 1 & 1 & 1 \end{pmatrix}$$

$$q_2^i = -\det \begin{pmatrix} x_0 & x_1 & x_i \\ y_0 & y_1 & y_i \\ 1 & 1 & 1 \end{pmatrix}$$

$$q_j^i = 0 \forall j = 3, 4, ...i-1, i+1, ...N-1 . \quad (3.2)$$

The invariants of the trajectory are defined as:

$$H^{N-3} = \{ q = (q_0, q_1, ..., q_{N-1})^T \in \mathbb{R}^{N-1} \}$$

$$Mq = (0, 0, 0)^T . \quad (3.3)$$
Figure 1. Visual illustrations of the motion trajectory images and their null space representation from CAVIAR data set.
3.2 Perturbation Analysis

3.2.1 2D Perturbation Analysis

Perturbation analysis is an important mathematical method which is widely used for analyzing the sensitivity of linear systems by adding a small term to the mathematical description of the exactly solvable problem. Let’s assume the noise matrix $Z$ as:

$$
\begin{bmatrix}
\epsilon_{x,0} & \epsilon_{x,1} & \cdots & \epsilon_{x,N-1} \\
\epsilon_{y,0} & \epsilon_{y,1} & \cdots & \epsilon_{y,N-1} \\
0 & 0 & \cdots & 0
\end{bmatrix},
$$

(3.4)

where $\epsilon_{x,i}, \epsilon_{y,i}$ are IID with zero mean and the variance $\delta^2$. Thus, the perturbed trajectory matrix $\tilde{M} = M + Z$ can be represented as:

$$
\begin{bmatrix}
x_0 + \epsilon_{x,0} & x_1 + \epsilon_{x,1} & \cdots & x_{N-1} + \epsilon_{x,N-1} \\
y_0 + \epsilon_{y,0} & y_1 + \epsilon_{y,1} & \cdots & y_{N-1} + \epsilon_{y,N-1} \\
1 & 1 & \cdots & 1
\end{bmatrix}.
$$

(3.5)

Let us derive the perturbed null operator with the first order perturbation.

$$
\tilde{q}_0^1 = -\det \begin{bmatrix}
x_1 + \epsilon_{x_1} & x_2 + \epsilon_{x_2} & x_i + \epsilon_{x_i} \\
y_1 + \epsilon_{y_1} & y_2 + \epsilon_{y_2} & y_i + \epsilon_{y_i} \\
1 & 1 & 1
\end{bmatrix}.
$$

(3.6)
Therefore, it is easy to obtain:

\[ \tilde{q}_i^i = q_i^i + \epsilon_{q_i^i}, \quad (3.7) \]

where \( \epsilon_{q_i^i} \) also has Gaussian distribution with zero mean and the variance \( [(x_i - x_1)^2 + (x_2 - x_1)^2 + (y_i - y_1)^2 + (y_i - y_2)^2] \delta^2 \), which is denoted as \( r_{12}^2 + r_{1i}^2 + r_{2i}^2 \delta^2 \) for simplicity. Similarly, we obtain:

\[ \tilde{q}_i^1 = q_i^1 + \epsilon_{q_i^1}, \quad \tilde{q}_i^2 = q_i^2 + \epsilon_{q_i^2}, \quad \tilde{q}_i^1 = q_i^1 + \epsilon_{q_i^1}, \quad (3.8) \]

where \( \epsilon_{q_i^1}, \epsilon_{q_i^2}, \epsilon_{q_i^1} \) satisfy:

\[ \epsilon_{q_i^1} \sim N(0, (r_{02}^2 + r_{01}^2 + r_{21}^2) \delta^2), \quad \epsilon_{q_i^2} \sim N(0, (r_{01}^2 + r_{02}^2 + r_{12}^2) \delta^2), \quad \epsilon_{q_i^1} \sim N(0, (r_{01}^2 + r_{02}^2 + r_{12}^2) \delta^2) \quad (3.9) \]

3.2.2 Discussion of The Error Ratio in 2D

Based on the perturbed null operator, it is desirable to know the ratio of the input error and the output error where the input error is referred to the error of the trajectory matrix and the output error is referred to the error of the null operator. Let us compute the expectation of the square of the Frobenius norm for the input error and the output error respectively. It can be shown that:

\[ E\|Z\|^2_F = 2N\delta^2 \quad (3.10) \]
Denoting the perturbed null operator matrix $\tilde{Q} = \begin{pmatrix} \tilde{q}_3^0 & \cdots & \tilde{q}_0^{N-1} \\ \tilde{q}_1^3 & \cdots & \tilde{q}_1^{N-1} \\ \vdots & \cdots & \vdots \\ \tilde{q}_{N-1}^3 & \cdots & \tilde{q}_{N-1}^{N-1} \end{pmatrix}$, $E\|Q - \tilde{Q}\|_F^2$ can be computed:

$$E\|Q - \tilde{Q}\|_F^2 = \sum_{i=3}^{N-1} \left[ E(\epsilon^2_{q_i}) + E(\epsilon^2_{q_i^1}) + E(\epsilon^2_{q_i^2}) \right] = 2\delta^2 \left[ (N - 3)(r_{01}^2 + r_{02}^2 + r_{12}^2) + \sum_{i=3}^{N-1} (r_{0i}^2 + r_{1i}^2 + r_{2i}^2) \right].$$

(3.11)

So the ratio of the output error and input error is:

$$\tau = \frac{E\|Q - \tilde{Q}\|_F^2}{E\|Z\|_F^2} = \frac{1}{N} \left[ (N - 3)(r_{01}^2 + r_{02}^2 + r_{12}^2) + \sum_{i=3}^{N-1} (r_{0i}^2 + r_{1i}^2 + r_{2i}^2) \right].$$

(3.12)

Therefore, it can be seen that the ratio only relies on the trajectory itself while independent of the noise. Minimizing $\tau$, we obtain:

$$x_i = \frac{x_0 + x_1 + x_2}{3}, y_i = \frac{y_0 + y_1 + y_2}{3}.$$  

(3.13)

It indicates that the centroid of the first three points gives the minimum value of the ratio of the output and the input error.
3.2.3 Discussion of the SNR in 2D

Besides the error ratio, SNR is widely used to evaluate robustness of the system. Let us derive the expression for the output SNR. Defining the power of the output signal as $\|Q\|_F^2$, SNR can be computed by $\Delta_{\text{SNR}} = \frac{\|Q\|_F^2}{E\|Q-Q\|_F^2}$ as:

$$
\Delta_{\text{SNR}} = \frac{A N}{2} \sum_{i=3}^{N-1} y_i^2 + B \sum_{i=3}^{N-1} x_i^2 + C \sum_{i=3}^{N-1} x_i y_i + D \sum_{i=3}^{N-1} x_i + E \sum_{i=3}^{N-1} y_i + F]/[2\delta^2(N-3) \sum_{j,k=0}^{2} r_{jk}^2 \\
+ \sum_{i=3}^{N-1} (r_{0i}^2 + r_{1i}^2 + r_{2i}^2)],
$$

(3.14)

where

$$
A = \sum_{j,k=0}^{2} (x_j - x_k)^2, \quad B = \sum_{j,k=0}^{2} (y_j - y_k)^2, \quad C = -2 \sum_{j,k=0}^{2} (x_j - x_k)(y_j - y_k),
$$

(3.15)

$$
D = 2 \sum_{j,k=0}^{2} (y_j - y_k)(x_j y_k - x_k y_j), \quad E = 2 \sum_{j,k=0}^{2} (x_j - x_k)(x_j y_k - x_k y_j),
$$

(3.16)

$$
F = (N-3)\left[ \sum_{j,k=0}^{2} (x_j y_k - x_k y_j)^2 + (x_1 y_2 - x_2 y_1 + x_0 y_1 - x_1 y_0 - x_0 y_2 + x_2 y_0)^2 \right]
$$

(3.17)

where $j \neq k$. It can be seen that (1) the critical points of SNR are trajectory-dependent; (2) SNR is invariant if each vector $v_i = (q_i^0, q_i^1, \ldots, q_i^{N-1})^T$ is normalized by the same constant.
3.2.4 3D Perturbation Analysis

In 3D case, let us assume the noisy data as \( \tilde{P}_i = (x_i + \epsilon_{x,i}, y_i + \epsilon_{y,i}, z_i + \epsilon_{z,i}) \), where \( \epsilon_{x,i}, \epsilon_{y,i}, \epsilon_{z,i} \) have zero mean and the variance \( \delta^2 \). Relying on the first order perturbation, we obtain the difference of the perturbed null operator \( \tilde{p}^{(i)}_0 \) and the true null operator:

\[
\tilde{p}^{(i)}_0 - p^{(i)}_0 = -\left[ \det \begin{pmatrix}
\epsilon_{x,1} & \epsilon_{x,2} & \epsilon_{x,3} & \epsilon_{x,i} \\
y_1 & y_2 & y_3 & y_i \\
z_1 & z_2 & z_3 & z_i \\
1 & 1 & 1 & 1
\end{pmatrix}
+ \det \begin{pmatrix}
x_1 & x_2 & x_3 & x_i \\
y_{y,1} & y_{y,2} & y_{y,3} & y_{y,i} \\
z_1 & z_2 & z_3 & z_i \\
1 & 1 & 1 & 1
\end{pmatrix}
+ \det \begin{pmatrix}
x_1 & x_2 & x_3 & x_4 \\
y_1 & y_2 & y_3 & y_4 \\
\epsilon_{z,1} & \epsilon_{z,2} & \epsilon_{z,3} & \epsilon_{z,i} \\
1 & 1 & 1 & 1
\end{pmatrix} \right].
\]  

(3.18)

\[
E \| \tilde{p}^{(i)}_0 - p^{(i)}_0 \|_F^2 = \sum_{j,k,m} \left[ (\det \begin{pmatrix}
y_j & y_k & z_m \\
z_j & z_k & z_m \\
1 & 1 & 1
\end{pmatrix})^2 + (\det \begin{pmatrix}
x_j & x_k & x_m \\
y_j & y_k & y_m \\
1 & 1 & 1
\end{pmatrix})^2 \right] \delta^2.
\]

(3.19)
where \( j = 1, 2; k = 2, 3; m = 3, i \) and \( j \neq k \neq m \). We evaluate the sensitivity of the system by the output noise produced from the unit input noise, namely, the error ratio \( \tau = \frac{E\|\tilde{Q} - Q\|^2_F}{E\|Z\|^2_F} \), where \( Z \) is the \( 4 \times N \) input noise matrix and the matrix \( \tilde{Q} = [\tilde{v}_4, \ldots, \tilde{v}_n] \), it is easy to show that

\[
E\|\tilde{Q} - Q\|^2_F = \left( (n - 4) \sum_{j,k,m} \left[ (\det \begin{pmatrix} x_j & x_k & x_m \\ y_j & y_k & y_m \\ 1 & 1 & 1 \end{pmatrix})^2 + (\det \begin{pmatrix} x_j & x_k & x_m \\ 1 & 1 & 1 \\ z_j & z_k & z_m \end{pmatrix})^2 \right] 
+ (\det \begin{pmatrix} 1 & 1 & 1 \\ y_j & y_k & y_m \\ z_j & z_k & z_m \end{pmatrix})^2 + 2 \sum_{i=4}^{n-1} \sum_{p,q} \left[ (\det \begin{pmatrix} x_p & x_q & x_l \\ y_p & y_q & y_l \\ 1 & 1 & 1 \end{pmatrix})^2 \right] \right) \delta^2 ,
\]

(3.20)

where \( j = 0, 1; k = 1, 2; m = 2, 3 \) and \( j \neq k \neq m \), \( p = 0, 1, 2; q = 1, 2, 3 \) and \( p \neq q \). So the error ratio can be computed as

\[
\tau = \frac{E\|\tilde{Q} - Q\|^2_F}{E\|Z\|^2_F} = \frac{E\|\tilde{Q} - Q\|^2_F}{3N\delta^2} .
\]

(3.21)

Similarly as the definition of SNR in 2D case, it is easy to obtain the exact expression of SNR in 3D case.
3.2.5 Bounds for the perturbation error

For both trajectories and images, using basic perturbation theorem as in (24), we obtain the normwise bound:

$$\frac{\|\tilde{Q} - Q\|}{\|Q\|} \leq \|M^{-1}Z\|,$$

(3.22)

where $\|\cdot\|$ denotes a matrix norm or a consistent vector norm. If we weaken the bound, we can relate it with the condition number of $M$.

$$\frac{\|\tilde{Q} - Q\|}{\|Q\|} \leq \|M^{-1}\|\|Z\| = \kappa(M) \frac{\|Z\|}{\|M\|},$$

(3.23)

where the condition number $\kappa(M) = \|M\|\|M^{-1}\|$. Since the left-hand side of the bound can be regarded as a relative error in $Q$. The factor $\frac{\|Z\|}{\|M\|} = \frac{\|\tilde{M} - M\|}{\|M\|}$ can likewise be regarded as a relative error in $\tilde{M}$. Thus the condition number $\kappa(M)$ implies how much the relative error in the matrix of the system $MQ = 0$ is magnified in the solution. With simple transformation,

$$\frac{\|Q\|}{\|Q - \tilde{Q}\|} \geq \frac{1}{\|M^{-1}Z\|}, \quad \frac{\|\tilde{Q} - Q\|}{\|Z\|} \leq \|M^{-1}\|\|Q\|.$$

(3.24)

In general, the inequalities (24) can be considered as the bound of the relative error for the multi-dimensional null space invariants. If in addition $\|M^{-1}Z\| < 1$, it is easy to obtain:

$$\frac{\|\tilde{Q} - Q\|}{\|Q\|} \leq \frac{\|M^{-1}Z\|}{1 - \|M^{-1}Z\|}.$$

(3.25)
The difficulty with normwise perturbation analysis is that it attempts to summarize a complicated situation by the relation between three numbers: the normwise relative error in $\tilde{Q}$, the condition number $\kappa(M)$ and the relative error in $\tilde{Q}$. We can do better if we are willing to compute the inverse of $M$:

$$|\tilde{Q} - Q| \leq |M^{-1}||Z||\tilde{Q}|.$$ (3.26)

Moreover, if for some consistent matrix norm $|||M^{-1}||Z|| < 1$, then $(I - |M^{-1}||Z||)^{-1}$ is nonnegative and

$$|\tilde{Q} - Q| \leq (I - |M^{-1}||Z||)^{-1}|M^{-1}||Z||Q|.$$ (3.27)

The bounds in (26), (27) which are referred as componentwise bounds (24) can be quite an improvement over normwise bounds. The superiority of eqs. (11) and (14) over the bounds in the (22)-(27) are two fold: (1) Compared to (11), although the inequality (22)-(27) gives a more compact form, they are not computationally feasible when the information of the noise matrix is not available; (2) instead of inequalities, eq. (14) provides a more straightforward formula to analyze the critical points and the convergence of SNR of null space invariants.

### 3.3 Sampling Strategy and Convergence Analysis

Given the perturbation of the null operators for motion trajectories, designing optimal sampling strategy is very important for the robustness of the system. In some scenario, such as the video sequences, uniform sampling is required to ensure the quality of the video. In
other cases, such as animal mobility experiments and GPS tracking, Poisson sampling is an important technique for obtaining the information.

### 3.3.1 Uniform sampling

Arbitrary trajectories in x and y directions can be represented as:

\[ x = f(t), y = g(t). \]  \hspace{1cm} \text{(3.28)}

Expanding in Maclaurin series, we obtain:

\[ f(t) = f(0) + f'(0)t + \ldots + \frac{f^{(n)}(0)}{n!}t^n, \quad g(t) = g(0) + g'(0)t + \ldots + \frac{g^{(n)}(0)}{n!}t^n. \]  \hspace{1cm} \text{(3.29)}

The distance between two arbitrary samples can be computed as:

\[ r_{kj}^2 = (x_k - x_j)^2 + (y_k - y_j)^2 = [f'(0)(t_j - t_k) + \ldots + \frac{f^{(n)}(0)}{n!}(t^n_j - t^n_k)]^2 + \frac{g^{(n)}(0)}{n!}(t^n_j - t^n_k)^2. \]  \hspace{1cm} \text{(3.30)}

With uniform sampling \( t_k = kT \), it is easy to obtain that \( \tau \) is equal to \( O(N^{2n}) \) which increases dramatically with the number of samples [33]. With regard to SNR, we have the following property:

**Property 1:** With uniform sampling \( t_k = kT \),

\[ \lim_{N \to \infty} \Delta_{SNR} = \frac{A(g^{(n)}(0))^2 + B(f^{(n)}(0))^2}{6\delta^2[(g^{(n)}(0))^2 + (f^{(n)}(0))^2]} + \frac{Cf^{(n)}(0)g^{(n)}(0)}{6\delta^2[(g^{(n)}(0))^2 + (f^{(n)}(0))^2]}, \]  \hspace{1cm} \text{(3.31)}
where A, B and C are defined in the equation (15).

**Remark:** Property 1 can be proved by showing that both $\|Q\|_F^2$ and $\|Q - \tilde{Q}\|_F^2$ are $O(N^{2n+1})$.

Thus, SNR is determined mainly by three factors: (1) The coordinates of the first three samples; (2) the values of the nth derivatives at the origin; and (3) the variance of the noise. The details of the proof can be found in Appendix II.

### 3.3.2 Poisson sampling

To guarantee the convergence of the error ratio, Poisson sampling is chosen which has the distribution for the sampling time $t_k$ as:

$$f(t_k) = \frac{\lambda^k t^{k-1} e^{-\lambda t}}{(k-1)!}.$$  \hspace{1cm} (3.32)

The nth moment of $t_k$ can be expressed as:

$$E(t_k^n) = \int_0^{+\infty} \frac{\lambda^k t^{k+n-1} e^{-\lambda t}}{(k-1)!} dt = \frac{(n+k-1)!}{\lambda^n (k-1)!}.$$  \hspace{1cm} (3.33)

**Property 2:** $\lambda = O(N)$ should be chosen for Poisson sampling to guarantee the convergence of $\tau$, where $N$ is the total number of samples.

**Property 3:** $\tau$ converges in mean sense given $\lambda = O(N)$. Specifically, for $\lambda = \frac{N}{\tau}$,

$$\lim_{N \to \infty} E(\tau) = 3 \sum_{k=1}^{2n} \frac{(a_k + b_k) T_k^k}{k+1},$$  \hspace{1cm} (3.34)
where $k$ is the index for Taylor series and for $a_k$ and $b_k$, if $k$ is odd,

$$a_k = \sum_{i=1}^{k/2} \frac{2g^{(i)}(0)g^{(k-i)}(0)}{i!(k-i)!}, \quad b_k = \sum_{i=1}^{k/2} \frac{2f^{(i)}(0)f^{(k-i)}(0)}{i!(k-i)!},$$

(3.35)

if $k$ is even,

$$a_k = \sum_{i=1}^{k-2} \frac{2g^{(i)}(0)g^{(k-i)}(0)}{i!(k-i)!} + \frac{(g^{(k/2)}(0))^2}{(k/2)!^2}, \quad b_k = \sum_{i=1}^{k-2} \frac{2f^{(i)}(0)f^{(k-i)}(0)}{i!(k-i)!} + \frac{(f^{(k/2)}(0))^2}{(k/2)!^2}.$$ (3.36)

The details of the proof can be found in Appendix I.

**Property 4:** If the trajectories are sampled with $\lambda = O(N)$, the variance of the error ratio converges to zero, namely,

$$\lim_{N \to \infty} \text{Var}(\tau) = 0.$$ (3.37)

**Remark:** In our framework, the density $\lambda$ corresponds to the average number of samples per unit-length; i.e. $\lambda = \frac{N}{T}$. The details of the the proof can be found in Appendix III.

### 3.4 View Invariant Feature Representation for Retrieval and Classification

For the objects with $m$ dimensional features, once we have generated the affine invariant representation provided by the null space operator, $\text{NSI}_{n \times (n-m-1)}$, where $m=2$ for motion trajectories, we can rely on numerous methods for indexing and classification. We choose a method for dimensionality-reduction and classification based on PCNSA (19). Notice that the term null space used in PCNSA is meant that the Approximate Null Space (ANS) used for
representation of each class is formed from the minimal eigenvectors within the class and thus minimizes the intra-class variance. However, this process is not intended to capture the null operator and is unrelated to the null space invariant proposed in Section III.

First, NSI is converted to \( P = n(n - m - 1) \) column vector \( Y_p \) which is assumed in class \( C_i \) and has Gaussian distribution as \( Y(Y \in C_i) \sim N(\mu_{full,i}, \Sigma_{full,i}) \), where \( \mu_{full,i} \) is the class conditional mean and \( \Sigma_{full,i} \) is the class conditional covariance matrix. To decrease the high dimensionality of NSI, we perform Principal Component Analysis (PCA), which removes the noise-only directions and retains the directions that yield large inter-class variance. PCA takes the \( L \) leading eigenvectors of covariance matrix, \( \Sigma_{full} \), of the entire data taken from all classes. The total scatter matrix, \( \Sigma_{full} \), can be written as \( \Sigma_{full} = \Sigma_{full,w} + \Sigma_{full,b} \) where \( \Sigma_{full,w} \) is within class covariance matrix and \( \Sigma_{full,b} \) between class covariance matrix i.e.

\[
\Sigma_{full,w} = \frac{1}{C} \sum_{i=1}^{C} \frac{1}{K} \sum_{k=1}^{K} (Y(i,k) - \mu_{full,i})(Y(i,k) - \mu_{full,i})^T,
\]

\[
\Sigma_{full,b} = \frac{1}{C} \sum_{i=1}^{C} (\mu_{full,i} - \mu_{full})(\mu_{full,i} - \mu_{full})^T,
\]

where \( i \) is for class index and \( k \) is for trajectory index in the class. It is assumed that there are \( C \) classes in the system and each class has \( K \) trajectories.
PCA gives the L-dimensional projection matrix \( \{W_{PCA}\}_{p \times L} \) and the projections into the PCA space are

\[
(X)_{L \times 1} = W_{PCA}^T(Y - \mu_{\text{full}}) \sim N(\mu_i, \Sigma_i),
\]

(3.40)

\[
(\mu_i)_{L \times 1} = W_{PCA}^T(\mu_{\text{full},i} - \mu_{\text{full}}), (\Sigma_i)_{L \times L} = W_{PCA}^T \Sigma_{\text{full},i} W_{PCA}
\]

(3.41)

After projections, in the PCA space PCNSA finds for each class i an \( M_i \) dimensional subspace along which the class’s intra-class variance is smallest. This subspace is referred to as the Approximate Null Space (ANS) denoted as \( N_i \) since the lowest eigenvalues’ corresponding eigenvectors are taken. That means we choose the lowest noise variance directions as for ANS.

**Assumptions:**

1. If \( \lambda_{\text{max},i} \) and \( \lambda_{\text{min},i} \) are the maximum and minimum eigenvalues of \( \Sigma_i \), there should be a threshold number such that \( \lambda_{\text{max},i}/\lambda_{\text{min},i} > \delta_1 \). This guarantees Approximate Null Space (ANS) for \( i \)th class.

2. There should be an another threshold number such that \( ||(\mu_i - \mu_j)^T e_i|| > \delta_2 ||\mu_i - \mu_j|| \), where \( e_i \) is any column of \( N_i \). This guarantees that any class i is linearly separable form other class j.

To get better solutions \( \delta_1 \times \delta_2 \) multiplication should be high i.e. \( \delta_1 = 10^7, \delta_2 = 10^{-4} \) can work.

**PCNSA Algorithm:**
1. **Obtain PCA Space:** Evaluate the total covariance matrix $\Sigma_{\text{full}}$, then apply PCA to the $\Sigma_{\text{full}}$ to find 

$$(W_{\text{PCA}})_{P \times L},$$
whose columns are the $L$ leading eigenvectors.

2. Project the data vectors, class means and class covariance matrices into the corresponding data vectors, class means, and class covariance matrices in the PCA space.

3. **Obtain ANS:** Find the approximate null space $(N_i)_{L \times M_i}$, for each class $i$ by choosing $M_i$ smallest eigenvalues' corresponding eigenvectors.

4. **Obtain Valid Classification Directions in ANS:** Say $N_i = (e_{i,1} | e_{i,2} | \ldots | e_{i,M_i})_{L \times M_i}$. If any direction, $e_i$ satisfies $|e_i^T(\mu_i - \mu_j)| > \delta_2\|\mu_i - \mu_j\|$, this direction is said valid direction and used to build valid ANS, $W_{\text{NSA},i}$.

5. **Classification:** PCNSA finds distances from a query trajectory/image to all classes

$$d_i(X) = \|W_{\text{NSA},i}(X - \mu_i)\|.$$ \hspace{1cm} (3.42)

We choose the smallest distance to a class for classification of $X$.

6. **Retrieval:** We compute the distance of the query trajectory or image to any other trajectory or image by $D(X_i, Y) = \|W_{\text{NSA},i}(X_i - Y)\|$, where $Y$ is the query trajectory or image.
3.5 ViewInvariant Motion Trajectories and Images Retrieval and Classification

3.5.1 Dataset

To implement and evaluate the proposed classification and retrieval system, we have used three data sets:

(1) The Australian Sign language (ASL) data set obtained from University of California at Irvine’s Knowledge Discovery in Databases archive (20). The trajectories in the data set are obtained by registration of the hand coordinates at each successive instant of time by using a Power Glove interfaced to the system. In our simulations, we used 40 different classes representing signing of 40 different words in the data set. Each class has 69 trajectories recorded at different instances.

(2) The CAVIAR Dataset: One of the most popular data sets used for video surveillance test is the Context Aware Vision using Image-based Active Recognition (CAVIAR) data set, which contains a number of video clips that were recorded while acting out different scenarios of interest, including multiple people walking, meeting with others, window shopping, entering and exiting shops, fighting and passing out, etc. This dataset provides the coordinates of the tracked objects as ground truth data, which can be used to determine motion trajectories from arbitrary moving cameras. In the experiment, to further demonstrate the view invariance nature of our framework, we populate the CAVIAR databases with 5 rotated versions of motion trajectories for each class. Typically, the database contains 20 classes with original trajectories and 5 rotated versions of each one at equal intervals in the -60 to +60 degrees range. (3) The
UMIST database: this database is available at http://images.ee.umist.ac.uk/danny/database.html. The UMIST Face Database consists of 564 images of 20 people. Each covering a range of poses from profile to frontal views. Subjects cover a range of race, sex and appearance.

3.5.2 View Invariance for Fixed Cameras From Unknown Views

In this subsection, we evaluate the performance for the classification and retrieval of motion trajectories with fixed cameras from unknown views. Namely, all the samples along the trajectories undergo the same global affine transformations. For all the simulations $\delta_1 = 10^7$, $\delta_2 = 10^{-4}$ as thresholds and $L = 32$ in PCNSA. Although we did not show here, we note that increasing L gives better results. Since in the real world, the trajectories in a class usually have different lengths, we normalize the length by taking 2D Fourier Transform of the trajectory matrix: $M = \begin{pmatrix} x_0 & x_1 & \ldots & x_{n-1} \\ y_0 & y_1 & \ldots & y_{n-1} \end{pmatrix}$ and choosing the largest 32 coefficients and then taking 2D Inverse Fourier Transform so that all the trajectories are of the same length. The visual illustrations of normalization with our approach for the trajectories taken from the class "come" in ASL dataset are provided in Fig.2(a). We have compared our approach with other normalization approaches such as upsampling and downsampling, the FFT and IFFT based normalization approach gives the better classification and retrieval performance since it utilizes the most information of the original data. Based on NSI, the $N \times (N-3)$ matrix $Q$ is converted into $n(n-3)$ column vector $Y$ which is considered as data samples. For retrieval problems, we compute the distance of the query trajectory to any other trajectory using PCNSA on NSI with the approach discussed in Section VI. This distance is then used to find $\alpha$ nearest trajectories,
where $\alpha$ is a user specified parameter. The visual examples of the top two retrieval and the query for the motions "chase" and "shopping and level" in retrieval are provided relying on the CAVIAR dataset in the first and the second row of Fig. 3 respectively. Fig.4(a) depicts accuracy of the proposed classification system versus number of classes. There are $K = 20$ trajectories in each class word. Simulation results show that our system preserves its efficiency even for higher number of different classes. Fig.4(b) depicts accuracy values versus increase in the number of trajectories within a class. There are $C = 20$ classes in the system. Simulation results show that our system performance deteriorates slightly for high number of trajectories in a class. This problem can be resolved by using a hierarchical representation, where we first separate all the trajectories in the system into a small number of classes, then repeatedly divide each class into smaller classes until each class has a sufficiently small number of trajectories. We can now use the null space representation for each class in the nested hierarchical tree structure to obtain a scalable representation whose performance is robust as the number of trajectories in the system increases. For graphical analysis, we plot two Precision Recall curves in Fig.4(c), where one is only with PCA on NSI which has the worse performance than our approach with PCNSA on NSI.

For noisy trajectories, relying on PCNSA, We plot the PR Curve for the classification of perturbed trajectories with the noise which has zero mean and the variance 0.5 in Fig.5(a). The PR curves for indexing and retrieval of true and noisy trajectories with the noise of zero mean and the variance 0.5 respectively for 20 classes are presented in Fig.5(b).
Figure 2. (a) Visual illustration for trajectories normalization for of the "come" class from ASL dataset. The "+" curves are original trajectories and ◦ curves are normalized trajectories. (b) Comparison of retrieval time with three approaches from ASL and CAVIAR dataset, X axis stands for the segmentation length, Y axis stands for the retrieval time (sec).

3.5.3 View Invariance for Arbitrary Moving Cameras

Compared to the situation of fixed cameras from unknown views where all the features points undergo the global affine transformations, the classification and retrieval problem is further compound when each of the feature points can undergo different affine transformations (eg. the cameras are moving arbitrary). In this case, computing null space of segmented trajectories yields higher accuracy since the orientations and the translations for adjacent points are very close, therefore they have more similar null space representation locally. With regard to that, the segmentation approaches come into the pictures. Typically, two approaches of classification
Figure 3. Visual illustration for retrieval results with 20 classes with motion trajectories from CAVIAR dataset for the motion events "chase" and "shopping and leave" for fixed cameras from unknown views.

Figure 4. (a) Accuracy values for the classification problem on increasing number of classes, (b) Accuracy values for the classification problem on increasing number of trajectories in a class. (c) The comparison with Precision-Recall Curve for retrieval with 20 classes for perfect trajectories.
and retrieval for motion events in the presence of arbitrary moving camera are proposed as follows:

1. **Overlapping segmentation:** Segmentation of overlapping sequences of k samples (with \( k \geq 4 \)) used to update the trajectory for the current samples, namely, we assume within the adjacent k samples, the points along the trajectory are rotated with the same angles with respect to the camera provided that the sampling frequency is fast enough.

2. **Non-overlapping segmentation:** Segmentation of non-overlapping sequences of k samples (with \( k \geq 4 \)) used to update the trajectory for all N samples.

We first segment the motion trajectories with equal lengths and subsequently compute the null space for segmented trajectories. Noticed that we can always normalize the length of the trajectory N to guarantee that \( \frac{N}{k} \) is an integer, by segmenting the trajectory with non-overlapping sequences of k samples, the null space invariant (NSI) matrix for each segmented trajectory is of dimension \( k \times (k - 3) \) and there are \( \frac{N}{k} \) segmented trajectories. So the NSI matrix is converted into \( k \times (k - 3) \times \frac{N}{k} = (k - 3)N \) column vector \( Y \) considered as data samples. For overlapping sequences of k samples, the null space invariant (NSI) matrix for each segmented trajectory is still of dimension \( k \times (k - 3) \) and there are \( N - k + 1 \) segmented trajectories. In this case, the NSI matrix is converted into \( k \times (k - 3) \times (N - k + 1) = k(k - 3)(N - k + 1) \) column vector \( Y \) considered as data samples. The computation complexity for covariance matrix of N dimensions is \( O(N^2) \). Therefore, given \( k \ll N \), for global trajectories, the computation complexity for the between covariance matrix in PCNSA is \( O(N^4) \). While by segmentation, the computation complexity for the between covariance matrix is reduced to \( O(N^2) \) for both
overlapping sequences and non-overlapping sequences.

Fig. 6 demonstrates the effectiveness of segmented approach in classification and retrieval of motion trajectories from moving cameras. The first row in Fig. 6 represents the query and the top three retrieval results in the motion event "entering the shop" based on the non-overlapping segmented trajectories from CAVIAR dataset. Specifically, we normalize each of the trajectories to the length 25 and segment each trajectory by the length of 5. We plot the corresponding first 16 nonzero null space invariants with global and segmented approaches in the second and the third rows. It can be seen that the null space invariants are very different in the second row. While they are very similar in the third row with the segmentation. The main reason is
that the similarity of the affine transformation for adjacent points yields the similarity of null space of the local feature points. Therefore, the null space invariants with segmented feature points is very discriminative for the classification and retrieval of video frames in the presence of arbitrary moving cameras.

We compare the performance of retrieval with overlapping segmentation and non-overlapping segmentation in Fig.7. Both of the two approaches obtain the same results for the top three retrieval for the motion event ”walking by the shop” with the segmentation length 5 while it has to be noted that the other retrieval results could be different. The null space representations with overlapping segmentation and non-overlapping segmentation are very close for the four video frames shown in the first row of Fig.7. Moreover, the null space representation with non-overlapping segmentation in the third row can be considered as the sub-sampling results for the ones with overlapping segmentation.

<table>
<thead>
<tr>
<th>k</th>
<th>ASL, lapping</th>
<th>ASL, Non-lapping</th>
<th>CAVIAR, lapping</th>
<th>CAVIAR, Non-lapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>0.857</td>
<td>0.638</td>
<td>0.832</td>
<td>0.756</td>
</tr>
<tr>
<td>5</td>
<td>0.912</td>
<td>0.651</td>
<td>0.875</td>
<td>0.791</td>
</tr>
<tr>
<td>7</td>
<td>0.883</td>
<td>0.621</td>
<td>0.903</td>
<td>0.761</td>
</tr>
<tr>
<td>8</td>
<td>0.793</td>
<td>0.604</td>
<td>0.813</td>
<td>0.724</td>
</tr>
</tbody>
</table>
Figure 6. The first row are the visual illustrations for the query and the top three retrieval of the video frames with overlapping segmented approaches in CAVIAR dataset, the second row are the corresponding first 16 non-zero null space invariants with global motion trajectories, the third row are the corresponding first 16 non-zero null space invariants with overlapping segmented motion trajectories.
Figure 7. The first row are the visual illustrations for the query and the top three retrieval of the video frames with overlapping segmented approaches in CAVIAR dataset, the second row are the corresponding first 16 non-zero null space invariants with overlapping segmented trajectories, the third row are the corresponding first null space invariants with non-overlapping segmented motion trajectories.
Figure 8. Three sample trajectories from ASL dataset. For each trajectory, another two versions of noisy trajectories with arbitrary moving cameras are also shown. Original trajectories are drawn in blue with the variance of noise 0.02.

To evaluate the robustness of our system, we set the variance of the noise 0.02 and the segmentation length 6 in the experiment. We conduct Poisson sampling to minimize the sensitivity for noisy trajectories. Three example of visual illustration of the noisy trajectory and another two affine versions from arbitrary moving cameras are provided in Fig.8 for ASL dataset. Fig.9 illustrates the example of the trajectory "all" and two affine versions with and without Poisson sampling with $\lambda = 0.8$. As it can be seen from Fig.10, the null space representations with Poisson sampling are more similar than the ones without sampling. Namely, Poisson sampling greatly attenuates the noisy effects. Fig.2(b) demonstrates the comparison of computation time for retrieval with three approaches: global trajectories without segmentation, overlapping, non-
Figure 9. Visual illustration for an example of three versions of the same noisy trajectory from the class “all” in ASL dataset with and without Poisson sampling ($\lambda = 0.8$) respectively. Typically, (a),(b),(c) are the representations without Poisson samplings; (d),(e),(f) are the representations with Poisson samplings.
Figure 10. Comparison of null space representation for three versions of the same noisy trajectory from the class "all" in ASL dataset with and without Poisson sampling (the first 16 nonzero elements).

Figure 11. The block diagram of the proposed classification and retrieval system.
overlapping segmented trajectories on the rotated versions respectively where Y axis represents the segmentation length. The global trajectories consumes the longest retrieval time among the three approaches while the non-overlapping segmentation requires the minimum retrieval time. Moreover, we compare the influence of the window size for segmentation. In Table II, overlapping segmentation outperforms non-overlapping segmentation in accuracy since it extracts more information and makes better approximation for arbitrary moving cameras and in fact the non-overlapping segmentation contains only part of information with respect to overlapping segmentation. Specifically, the most accurate classification results for overlapping and non-overlapping segmentation are obtained respectively for ASL data set when $k = 5$. For CAVIAR data set, when $k$ is equal to 7 and 5, the most accurate classification results for overlapping and non-overlapping segmentation are detected. In addition, the segmented trajectories play a more important role in the following aspects: 1) due to the object being temporarily out of the field of view, or trackers losing the track of object for a while, only a part of motion trajectories are available; 2) Substantially reduce the high dimensionality of the
Figure 13. (a) Feature points extraction for the 1st and 6th frames based on SIFT in the UMIST face database, (b) Accuracy of classification with increasing number of classes, (c) Accuracy for classification problems with the increase in the number of images within a class, (d) Precision-Recall metric by using PCNSA on NSI for original and noisy images.
null space invariants. To summarize, the block diagram for the proposed system is presented in Fig.10.

3.5.4 Application in Facial Recognition

In this subsection, we further demonstrate the application of our systems in face recognition. We use scale invariant feature transform (SIFT) to obtain feature points based on SIFT algorithm in (21). SIFT extracts and connects feature points in images which are invariant to image scale, rotation and changes in illumination. Since the feature points of images in a class may have different numbers, we normalize the length of the vector formed by feature points by selecting the minimum number of feature points from all the images.

Traditionally, the 2D feature points in images and 3D coordinates in objects are associated with $3 \times 4$ projection matrix and many research work focus on designing projection matrix (22), which incorporates three coordinate systems: camera, image and world. While our retrieval and classification system relying on null space invariants does not require any information of the projection matrix. The applications of our system are twofold: 1) retrieval and classification for 2D images with unknown views 2) retrieval and classification for 3D images with unknown translation and rotation. The proposed classification and retrieval system is implemented and evaluated using the UMIST database (23) as in Fig.12, which consists of facial images of 20 people, namely, 20 different classes. In each class, we choose 24 images, captured from different poses. We extract 36 feature points from each image. In Fig.13 (a), an example of feature points extraction relying on SIFT are illustrated. These feature points are transformed into null space invariant, NSI, which is invariant to affine transformation since it preserves the values against
rotation and transformation. For all the simulations, we have chosen random noise with zero mean and variance $\delta_1^2 = 0.1$ for noisy image data, $\delta_2 = 5 \times 10^{-5}$ and 35 largest eigenvectors in obtaining the PCA space. Fig.13 (b) depicts accuracy of the proposed classification system versus number of classes for perfect images and noisy images. Simulation shows that our systems preserves its efficiency and robustness even with large number of classes and noisy data. Specifically, our approach is still valid and efficient even when the SNR decreases to 5dB.

Fig.13 (c) depicts accuracy values versus increase in the number of images in a class for perfect images and noisy images. Simulation results show that the performance of our system deteriorates slightly with the increase in the number of images within a class. Specifically, the performance deteriorates slightly when the difference of angles in the images taken from the object are larger than 45 degrees since there are fewer feature points to obtain. This problem can be solved by using recursive partition of the data sets. We divide the data set into smaller groups until we can extract enough and consistent feature points. In the experiment, the recursive partition stops when the difference of angles for two images are smaller than 30 degrees.

Fig.13 (d) shows Precision vs Recall curves for indexing and retrieval problem for 18 different classes of images, each class having 20 images. We compute the distance of the query image to any other image based on the approach discussed in Section V. This distance is then used to find $\gamma$ nearest images, where $\gamma$ is a user specified parameter. There are 3 curves in figures. One corresponds to directly using PCA on NSI for retrieval, which has the worst performance.
The other two curves correspond to using PCNSA on NSI for perfect and noisy data, both of which are much superior compared to the curve without using PCNSA.

### 3.5.5 Comparison with Traditional Approaches

In this section, we compare the performance of our approach with traditional approaches. For motion trajectories, curvature scale space (CSS) based affine-invariant trajectory retrieval as (32) has been proposed. The details of the process are shown in (32). The main superiorities of our null space invariant over the curvature scale space based approach are summarized as follows:

1. The CSS based approach is only approximately view-invariant in the situation for fixed cameras with unknown views, while our approach is perfect view-invariant in that situation.

2. Our approach can solve the classification and retrieval problem for moving cameras while the CSS based approach does not work in this case.

3. The CSS based approach is sensitive to noise, while null space invariants are robustness to noisy dataset.

4. Our approach can save significant computation time compared to curvature scale space based approaches.

The comparison of PR curve for our approach and curvature based approach for motion trajectory retrieval for perfect trajectories by fixed cameras with unknown views is provided in Fig.14(a). The curvature scale based approach performs poorly at high values of recall. For
Figure 14. (a) The comparison of PR curve for our approach and curvature based approach for motion trajectory retrieval for perfect trajectories, (b) The comparison of PR curve for our approach and CSS-based approach in retrieval for noisy trajectories.

the comparison with noisy dataset, we implement the classification and retrieval with the variance of the noisy 0.1 and the results are shown in Fig.14(b).
CHAPTER 4

TENSOR NULL-SPACE INVARIANTS

4.1 Introduction

Among affine view-invariance systems, majority of them represent affine view-invariance in a single dimension (18)(1), thus limiting the system to only single object motion based queries and single dimension affine view-invariance. In many applications, it is not only the individual movement of an object that is of interest, but also the motion patterns that emerge while considering synchronized or overlapped movements of multiple objects. For example, in sports video analysis, we are often interested in a group activity involving motion activity of multiple players, rather than the activity of each individual player. Moreover, due to camera movement, same motion trajectory has completely different representations from different viewing angles. Therefore, a highly efficient classification and retrieval system which is invariant to multidimensional affine transformations is highly desirable.

In this chapter, we propose a novel fundamental mathematical framework for tensor null space invariants and use this framework for the important application of view-invariant classification and retrieval of motion events involving multiple motion trajectories.\(^1\) Our main contributions are as follows: (i) we introduce tensor null space invariants (TNSI) which are perfect affine invariants in a multi-dimensional space, (ii) we demonstrate robustness and superiority of TNSI

\(^1\)Tensor representation in a finite-dimensional space is often referred to as multilinear algebra.
as a powerful tool in classification and retrieval of high-dimensional data over traditional approaches.

4.2 Tensor Null Space Invariants

Let us denote the tensor $A \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_{N-1} \times I_N}$ as the multi-dimensional data. Elements of $A$ are denoted as $a_{i_1, i_2, \ldots, i_N}$. As in (25), a generalization of the product of two matrices is the product of a tensor and a matrix. The mode-$n$ product of a tensor $A \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_{N-1} \times I_N}$ by a matrix $U \in \mathbb{R}^{I_n \times J_n}$, denoted by $A \times_n U$, is a tensor $B \in \mathbb{R}^{I_1 \times \cdots \times I_{n-1} \times J_n \times I_{n+1} \times \cdots \times I_N}$ whose entries are:

$$ (A \times_n U)_{i_1 \cdots i_{n-1} i_n i_{n+1} \cdots i_N} = \sum_{i_n} a_{i_1 \cdots i_{n-1} i_n i_{n+1} \cdots i_N} u_{i_n j_n} \quad (4.1) $$

The mode-$n$ product $B = A \times_n U$ can be computed via the matrix multiplication $B_{(n)} = UA_{(n)}$, followed by a re-tensorization to undo the mode-$n$ flattening.

As in (18), let $Q_i = (x_i, y_i)$ be a single 2-D point, $i = 0, 1, \ldots, n - 1$, among $n$ ordered nonlinear points in $\mathbb{R}^2$. Consider the following arrangement of the $n$ 2-D points in a $3 \times n$ matrix $M$:

$$ M = \begin{pmatrix} x_0 & x_1 & \cdots & x_{n-1} \\ y_0 & y_1 & \cdots & y_{n-1} \\ 1 & 1 & \cdots & 1 \end{pmatrix} \quad (4.2) $$
the null space $H^{n-3}$ can be represented as:

$$H^{n-3} = \{q = (q_0, q_1, \ldots, q_{n-1})^T, \text{i.e.} Mq = (0, 0, 0)^T\} \quad (4.3)$$

Applying the affine transformation $T$ on the matrix $M_1 = TM$, the null spaces of $M_1$ and $M$ are identical as described in (18). Similarly, applying the affine transformation $T_m, T_n$ on the $m$th, $n$th unfolding of the multi-dimensional data $M$, respectively, if the resulting tensor null space $Q$ is invariant in both dimensions, the it is referred to as mode-$m,n$ invariant. Let us derive the mathematical formulation of the mode-1,2,3 invariant tensor $Q$ for three dimensional data $M \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. To be rotation invariant, we have:

$$M_{(1)} \times Q_{(3)} = 0, M_{(2)} \times Q_{(2)} = 0, M_{(3)} \times Q_{(1)} = 0, \quad (4.4)$$

where $M_{(1)}, M_{(2)}, M_{(3)}$ are the unfolding of the three order tensor $M$ into matrices with the dimension $I_2I_3 \times I_1$, $I_1 \times I_3I_2$ and $I_1 \times I_2I_3$, respectively, and $Q_{(3)}, Q_{(2)}, Q_{(1)}$ are the corresponding unfoldings of the tensor $Q$. On the right side of (4), 0 represents the corresponding null tensor.

Let us assume the translation vector as $T_i = [t_i, \ldots, t_i]$, the translation matrix as $T = [T_1, \ldots, T_N]'$.

For example, for motion trajectories, $T = [T_1, T_2]'$, where $T_1$ represents the shift of all the coordinates in x dimension and $T_2$ represents the shift of all the coordinates in y dimension. Now we shall derive the condition on the TNSI $Q$ to guarantee the invariance of translation for tensor $M$. To derive the translation in the dimension $I_n$, we should unfold the tensor $M$ into matrix $M_{[n]} = [m_1, \ldots, m_N]'$ with the dimension $I_1I_2 \times \ldots$
In \( I_{n-1}I_{n+1} \ldots I_N \times I_n \). Assuming \( R \) as rotation matrix, according to the definition of tensor null space invariants:

\[
(RM_{(n)} + T)Q = [Rm_1 + T_1, ..., Rm_N + T_N]'Q = [Rm_1, ..., Rm_N]'Q + [T_1, ..., T_N]'Q = 0, \tag{4.5}
\]

Due to invariance of rotation, \([Rm_1, ..., Rm_N]'Q = 0\). Thus,

\[
[T_1, ..., T_N]'Q = [T_1Q, ..., T_NQ]' = [\sum t_1q_1, ..., \sum t_Nq_N] = [t_1, ..., t_N]\sum q_i = 0 \tag{4.6}
\]

Since \([t_1, ..., t_N]\) can be arbitrary, the condition for the equation (6) to hold true is that each column of the unfolding of TNSI should sum up to zero, namely, \(\sum q_i = 0\). Therefore, we obtain the condition for invariance of translation for the mode-1,2,3 invariant tensor \(Q\):

\[
\sum Q_{(1)} = 0, \sum Q_{(2)} = 0, \sum Q_{(3)} = 0 \tag{4.7}
\]

Combining equations (3.4) and (3.7), we can solve the TNSI \(Q\) subject to the mode-1,2,3 affine view-invariant. If the order of the tensor \(M\) is 2, it is easy to show that the condition boils down to the Stiller’s one dimensional null space invariants (18). It is also easy to extend the result to the case of the Nth order tensor with mode-1, ..., \(I_k\) affine view-invariant.
4.3 Classification and Retrieval Algorithms

We align each trajectory as two rows in a matrix according to x and y coordinates, and the number of rows of a matrix is set to be twice the number of the objects in the motion event under analysis.

$$M = (M_{i,j})_{i=1,2,...,2J; j=1,2,...,p}, \quad (4.8)$$

In the above equation, $p$ denotes the temporal length of normalized trajectories, $J$ represents the number of trajectories within one motion event. Finally, multiple trajectory matrices are aligned in the direction orthogonal to the plane spanned by them, and form a three dimensional matrix, or tensor. We refer to it as *Motion Event Tensor* $T$. as shown in Fig.15.

$$T = (T_{i,j,k})_{i=1,2,...,2J; j=1,2,...,p; k=1,2,...,K}, \quad (4.9)$$

where $K$ is the number of motion event samples (trajectory video clips). Once we have generated the affine invariant representation provided by the tensor null space operator, TNSI, we can rely on numerous methods for indexing and classification. We choose a method for dimensionality-reduction and classification based on PCNSA (19). Notice that the term null space used in PCNSA implies that the Approximate Null Space (ANS) used for the representation of each class is formed from the minimal eigenvectors within the class, and thus minimizes the intra-class variance. However, this process is not intended to capture the null operator and is unrelated to TNSI proposed in Section 2.
First, TNSI is converted to \( H = \prod_{i=1}^{n+1} W_i \) column vector \( Y_p \) which is assumed in class \( C_i \) and has Gaussian distribution as \( Y[Y \in C_i] \sim N(\mu_{\text{full},i}, \Sigma_{\text{full},i}) \), where \( \mu_{\text{full},i} \) is the class conditional mean and \( \Sigma_{\text{full},i} \) is the class conditional covariance matrix. To decrease the high dimensionality, we perform Principal Component Analysis (PCA), which removes the noise-only directions and retain the directions that yield large inter-class variance. PCA takes the \( L \) leading eigenvectors of covariance matrix, \( \Sigma_{\text{full}} \), of the entire data taken from all classes. The total scatter matrix, \( \Sigma_{\text{full}} \), can be written \( \Sigma_{\text{full}} = \Sigma_{\text{full},w} + \Sigma_{\text{full},b} \) where \( \Sigma_{\text{full},w} \) is within class covariance matrix and \( \Sigma_{\text{full},b} \) between class covariance matrix:

\[
\Sigma_{\text{full},w} = \frac{1}{CK} \sum_{i=1}^{C} \sum_{k=1}^{K} (Y_{i,k} - \mu_{\text{full},i})(Y_{i,k} - \mu_{\text{full},i})^T, \\
\Sigma_{\text{full},b} = \frac{1}{C} \sum_{i=1}^{C} (\mu_{\text{full},i} - \mu_{\text{full}})(\mu_{\text{full},i} - \mu_{\text{full}})^T,
\]

(4.10) (4.11)

where \( i \) is for class index and \( k \) is for motion event tensor index in the class. It is assumed that there are \( C \) classes in the system and each class has \( K \) tensors.

PCA gives the \( L \)-dimensional projection matrix \( (W_{PCA})_{p \times L} \). After projections, in the PCA space PCNSA finds for each class \( i \) an \( M_i \) dimensional subspace along which the class’s intra-class variance is smallest. This subspace is referred to as the Approximate Null Space (ANS) denoted as \( N_i \) since the lowest eigenvalues’ corresponding eigenvectors are taken. That means we choose the lowest noise variance directions as for ANS. **PCNSA Algorithm:**

1. **Obtain PCA Space:** Evaluate the total covariance matrix \( \Sigma_{\text{full}} \), then apply PCA to \( \Sigma_{\text{full}} \) to find
$W_{\text{PCA}}$, whose columns are the $L$ leading eigenvectors. Project the data vectors, class means and class covariance matrices into the corresponding data vectors, class means, and class covariance matrices in the PCA space.

2. Obtain ANS and Valid Classification Directions in ANS: Find the approximate null space $(N_i)_{L \times M_i}$, for each class $i$ by choosing $M_i$ smallest eigenvalues’ corresponding eigenvectors. $N_i = (e_{i,1} | e_{i,2} | \ldots | e_{i,M_i})_{L \times M_i}$. If $e_i$ satisfies $|(\mu_i - \mu_j)^T e_i| > \delta_2 \|\mu_i - \mu_j\|$, this direction is valid and used to build valid ANS, $W_{\text{NSA},i}$.

3. Classification: PCNSA finds distances from a query tensor to all classes and minimum distance to a class is chosen for classification of $X$.

$$d_i(X) = \|W_{\text{NSA},i}(X - \mu_i)\|. \quad (4.12)$$

4.4 Simulation Results

In order to implement and evaluate the proposed classification and retrieval system, we have used trajectories from the Australian Sign language (ASL) data set obtained from University of California at Irvine’s Knowledge Discovery in Databases (UCI-KDD) archive (20). The trajectories in the data set are obtained by registration of the hand coordinates at each successive instant of time by using a Power Glove interfaced to the system. In our simulations, we have used 40 different classes representing signing of 40 different words in the data set. Each class has 69 trajectories recorded at different instances.

Since in real life trajectories may have different lengths, we normalize the length by taking
the Fourier Transform and choosing the biggest $n=18$ coefficients and then taking the Inverse Fourier Transform so that all the trajectories are of size 32 before invariant matrix calculations.

We form the motion event tensor $\mathbf{T}$ by randomly selecting the trajectories from the specific class and setting $J=3$, $P=18$, $K=20$, namely, each tensor $\mathbf{T} \in \mathbb{R}^{6 \times 18 \times 20}$ contains $3 \times 20 = 60$ motion trajectories, each trajectory has 18 samples and totally there are 20 video clips. According to the definition of TNSI, applying affine transformation to the unfolding matrix $\mathbf{T}_{(1)}$ with the dimension $2J \times PK$ indicates rotation and translation of each trajectory at all video clips to the same amount. Applying affine transformation to the unfolding matrix $\mathbf{T}_{(3)}$ with the dimension $K \times 2PJ$ indicates rotation and translation of the same trajectory at each video clip independently. Therefore, we compute the two dimensional invariants with $\delta_2 = 10^{-4}$ as thresholds and $L = 50$ in PCNSA. Fig. 16 (a) depicts the accuracy of the proposed classification
Figure 16. Accuracy for tensor (multiple motion trajectories) classification (a) with an increasing number of classes and (b) with an increasing number of tensors within a class. System versus the number of classes. There are 20 tensors in each class. Simulation results show that our system preserves its efficiency even for higher number of different classes. Fig. 16 (b) depicts accuracy values versus increase in the number of tensors within a class. There are 20 classes in the system.
CHAPTER 5

LOCALIZED NULL-SPACE INVARIANTS

Due to advances in multimedia technology and the onset of popular online multimedia archives and social networks in recent years, there has been an increased interest in video event/object classification and retrieval (16)(29). However, several fundamental challenges in video classification and retrieval systems must be addressed before such systems can be employed in critical applications: Firstly, one of the main challenges is that events/objects captured from different viewpoints or moving camera sensors lead to completely different representations (31). Secondly, a fundamental obstacle in video classification and retrieval systems is that the indexed video representation must often be adjusted to cope with partial or limited information during the learning/indexing phase and query time. The information may be incomplete due to numerous issues such as missing features or objects occlusions (30), partial sensor modalities and limited exposure time due to video acquisition, etc. Finally, the learning/indexing phase is usually performed by dynamically updating the representation of the existing dataset. Efficient updating procedures for the representation of dynamic video databases are essential to the performance of video classification and retrieval systems. For example, in face recognition, at times only partial facial information from a different viewpoint is available either during indexing/learning or at query time. Similarly, in motion events, trajectory data is often missing segments due to occlusions. Another example arises when events in the database consist of, for instance, two interacting objects, whereas the query contains three interacting objects.
Moreover, in both face recognition and motion trajectory analysis, the visual information representation in the database must be updated in response to the integration of new visual data. Such scenarios require dynamic splitting and merging of the data representation. Executing such operations by computing modified representations from scratch using raw data is infeasible. These challenges require a representation space that is invariant to camera viewpoint and movement (e.g. affine transformation) that can be dynamically updated (i.e. merged or split) to perform classification and retrieval in response to augmentation of the database or the availability of partial and limited information.

In (16), two different affine invariant representations for motion trajectories have been used for trajectory recognition and classification, namely Curvature Scale Space (CSS) and Centroid Distance Functions (CDF). However, these techniques are view invariant only when the camera motions are small. A new invariant algorithm for the Structure From Motion (SFM) problem has been proposed in (17). It relies on the invariant property of group action on a vector space to eliminate the camera pose parameters in the calculations. This approach yields robust solutions for SFM. Although the invariant algorithm presented has the potential to be used effectively for recognition and classification problems, the requirement for solving the invariant equations using the Levenberg-Marquardt algorithm introduces a high computational complexity. In (26)(2), a robust view-invariant classification and retrieval system relying on null-space invariants (18) has been presented and demonstrated to provide a very effective means for the representation of image and video data captured from an unknown camera viewpoint or a mov-
Methods for merging and splitting feature spaces such as eigenspace and tensor subspace exist (33)(27)(28). However, these methods do not explore the view-invariance properties of null spaces. The design for the null space representation presented in (26) is not feasible for partial queries and dynamical updatings.

In the future, we provide a novel method for classification and retrieval of events or objects that is invariant to camera orientation and movement and allows for efficient merging and splitting of the database representation when only partial or limited information is available or when the visual archive is updated. Specifically, we propose a new Localized Null Space representation and present efficient updating and downdating techniques for dynamically adjusting the Localized Null Space. The resulting representation yields a much faster method for dynamic updating of the localized null space representation than the alternative of recomputing all of the elements used in the existing null space representation from raw data.

5.1 Localized Null Space based Representation

For the sake of completeness, in the following we first briefly overview the traditional Null Space. In (26), Null Space Invariant (NSI) of a feature matrix (each row corresponds to a single feature vector) has been introduced as a new and powerful affine invariant space to be used for feature representation. A fundamental set of 2-D affine invariants for an ordered set of \( n \) points in \( \mathbb{R}^2 \) (not all colinear) is expressed as an \( n - 3 \) dimensional subspace, \( \mathbb{H}^{n-3} \), of \( \mathbb{R}^{n-1} \),
which yields a point in the $2n - 6$ dimensional Grassmannian $\text{Gr}_R(n - 3, n - 1)$, a manifold of dimension $2n - 6$. Consider the $3 \times N$ matrix $M$ as:

$$
M = \begin{pmatrix}
  u_0 & u_1 & ... & u_{N-1} \\
  v_0 & v_1 & ... & v_{N-1} \\
  1 & 1 & ... & 1
\end{pmatrix}.
$$

(5.1)

The traditional null space (18) $H^{N-3}$ is spanned by the vector $q^{(i)} = (q_0^{(i)}, q_1^{(i)}, ..., q_{N-1}^{(i)})^T$, $i=3,4, ..., N-1$, where $i$ is the column index,

$$
q_0^{(i)} = -\det\begin{pmatrix}
  u_1 & u_2 & u_i \\
  v_1 & v_2 & v_i \\
  1 & 1 & 1
\end{pmatrix},
$$

$$
q_1^{(i)} = \det\begin{pmatrix}
  u_0 & u_2 & u_i \\
  v_0 & v_2 & v_i \\
  1 & 1 & 1
\end{pmatrix},
$$

$$
q_2^{(i)} = -\det\begin{pmatrix}
  u_0 & u_1 & u_i \\
  v_0 & v_1 & v_i \\
  1 & 1 & 1
\end{pmatrix},
$$

$$
q_i^{(i)} = \det\begin{pmatrix}
  u_0 & u_1 & u_2 \\
  v_0 & v_1 & v_2 \\
  1 & 1 & 1
\end{pmatrix}.
$$
\(q_j^{(i)} = 0 \forall j \in [3, N - 1], j \neq i\). \hspace{1cm} (5.2)

The invariants of the features of the points are defined as:

\[
H_{N-3} = \{q^{(i)} = (q_0^{(i)}, q_1^{(i)}, ..., q_{N-1}^{(i)})^T \in \mathbb{R}^{N-1} \}
\]

\[
Mq^{(i)} = (0, 0, 0)^T.
\] \hspace{1cm} (5.3)

When an input query consists of partial information or the database is to be augmented with new data, the above described traditional Null Space based representation is not feasible, as it requires access to the raw data to compute any arbitrary subspace or to update the existing space. As shown in Fig. ??, the traditional Null Space of segmented partial trajectory is very different from the Null Space of the original object.

To classify and recognize events or objects in the presence of affine transformations when the input contains only partial information, we propose a novel Localized Null Space. Note that in equation (2), each vector \(q^{(i)}\) always contains the information of the coordinates \((u_0, v_0), (u_1, v_1), (u_2, v_2)\), which we refer to as key points in NSI. However, it is not necessary to always utilize the first three points as key points as long as the invariants satisfy the condition in the underdetermined equation (3) and each vector is independent. Instead of utilizing the first three points as key points for all the vectors as in the traditional Null Space, the proposed Localized Null Space divides the raw feature vector into multiple non-overlapping
segments and computes the Null Space for each segment using only local key points. Without
loss of generality, let us assume the two non-overlapping segmented feature matrices are:

\[
W_1 = \begin{pmatrix}
  u_0 & u_1 & \ldots & u_{K-1} \\
  v_0 & v_1 & \ldots & v_{K-1} \\
  1 & 1 & \ldots & 1
\end{pmatrix},
\]

\[W_2 = \begin{pmatrix}
  u_K & u_{K+1} & \ldots & u_{N-1} \\
  v_K & v_{K+1} & \ldots & v_{N-1} \\
  1 & 1 & \ldots & 1
\end{pmatrix},
\]

(5.4)

(5.5)

**Property 1:** Localized null space invariants are computed as follows (3): For \( i \in [3, K+2] \),

\[
q_0^{(i)} = -\det \begin{pmatrix}
  u_1 & u_2 & u_i \\
  u_1 & u_2 & u_i \\
  1 & 1 & 1
\end{pmatrix},
\]

\[
q_1^{(i)} = \det \begin{pmatrix}
  u_0 & u_2 & u_i \\
  u_0 & u_2 & v_i \\
  1 & 1 & 1
\end{pmatrix},
\]

\[
q_2^{(i)} = -\det \begin{pmatrix}
  u_0 & u_1 & u_i \\
  v_0 & v_1 & u_i \\
  1 & 1 & 1
\end{pmatrix},
\]
\[
q_i^{(i)} = \det \begin{pmatrix} u_0 & u_1 & u_2 \\ v_0 & v_1 & v_2 \\ 1 & 1 & 1 \end{pmatrix}
\]
\[
q_{i}^{(i)} = 0 \forall j = 3, \ldots, i-1, i+1, \ldots, N-1. \tag{5.6}
\]

For \( i \in [K + 3, N - 1] \),

\[
q_K^{(i)} = -\det \begin{pmatrix} u_{K+1} & u_{K+2} & u_i \\ v_{K+1} & v_{K+2} & v_i \\ 1 & 1 & 1 \end{pmatrix},
\]

\[
q_{K+1}^{(i)} = \det \begin{pmatrix} u_{K} & u_{K+2} & u_i \\ v_{K} & v_{K+2} & v_i \\ 1 & 1 & 1 \end{pmatrix},
\]

\[
q_{K+2}^{(i)} = -\det \begin{pmatrix} u_{K} & u_{K+1} & u_i \\ v_{K} & v_{K+1} & v_i \\ 1 & 1 & 1 \end{pmatrix},
\]

\[
q_i^{(i)} = \det \begin{pmatrix} u_{K} & u_{K+1} & u_{K+2} \\ v_{K} & v_{K+1} & v_{K+2} \\ 1 & 1 & 1 \end{pmatrix},
\]

\[
q_{i}^{(i)} = 0 \forall j = 0, \ldots, K-1, K + 3, \ldots, i-1, i+1, \ldots, N-1. \tag{5.7}
\]
Remark: It is easy to verify that Localized Null Space (LNS) in equations (6) and (7) satisfy equation (3). Note that the formulation of LNS is not unique and it depends on the selection of key points and the length of segments. It can be easily extended to LNS for \( n \) feature vector segments.

The benefits of the LNS are mainly two fold: (1) the Localized Null Space can be viewed as consisting of multiple subspaces and therefore can be dynamically split to answer partial queries, (2) the Localized Null Space concept can be applied to merge several sub Null Spaces into one integrated Null Space. This allows dynamic updating without re-computing the Null Space from raw data. Fig. Figure 17 illustrates that Localized Null Space of a partial object matches exactly with the corresponding portion of the Null Space of the original object. (3) the Localized Null Space has almost the same computational complexity as traditional Null Space. Namely, it is computationally very efficient.

In the following sections we develop efficient merging and splitting procedures to facilitate dynamic updating of the proposed Localized Null Space.

5.2 Splitting Procedures for LNS

5.2.1 Deterministic Splitting

In deterministic splitting, we assume that the lengths of the segments and the key points in each segments are known to users. In the other words, the correspondence of the partial object with the data in the archive is known to the user. Therefore, the Localized Null Space Invariants can be derived. Figure 18 demonstrates the structure of the Localized Null Space
Figure 17. Visual illustrations of the original trajectory and part of the rotated motion trajectory with identical localized null space representation from CAVIAR data set.
5.2.2 Random Splitting

In some scenarios the length of each partial feature vector and the positions of key points with respect to the full feature vector may not be available to the user. In this section we present an optimal splitting approach and key points selection for random partial feature vectors.

5.2.2.1 Optimal Segmentation

Let us assume that the length of feature vectors is $L$ (the number of columns in the feature matrices) and the number of segments is $N$. The lengths of $N$ segments are denoted by
L_1, \ldots, L_N vector. Note that the least length for each segment to guarantee the existence of Null Space is 4, namely, L_i satisfies:

\[ L_i \geq 4, \sum_{i=1}^{N} L_i = L. \]  \hspace{1cm} (5.8)

The N − 1 segmentation points w_i are independent random variables with the distribution P(w_i), 0 \leq w_i \leq L. Also, notice that P(w_i) is not necessary to be identical, the distribution of L_i can be computed as follows:

\[ P(L_1) = P(w_1). \]  \hspace{1cm} (5.9)

For i ∈ [2, N − 1]

\[
P(L_i = j) = \sum_{k=4(i-1)}^{L-4(N-i+1)} P(w_i = k + j, w_{i-1} = k)
\]

\[ = \sum_{k=4(i-1)}^{L-4(N-i+1)} P(w_i = k + j)P(w_{i-1} = k) \]  \hspace{1cm} (5.10)

\[ L_N = L - \sum_{i=1}^{N-1} L_i, E(L_N) = L - \sum_{i=1}^{N-1} E(L_i). \]  \hspace{1cm} (5.11)

Assuming the optimal segmentation lengths as K_1, \ldots, K_N, the distortion for random splitting can be represented as
The problem is equivalent to minimizing the distortion,

\[ \min_{k_i} D, \; \text{st} \sum_{i=1}^{N} k_i = L. \]  \hspace{1cm} (5.13)

Using the method of Lagrange multipliers, the MMSE estimator of the length of each segment for random splitting can be determined.

**Property 2:** Given a query of length $L$ consisting of $N$ segments, the optimal length for each segment and minimum distortion can be represented as (?):

\[ k_i = E(L_i), \forall 1 \leq i \leq N \] \hspace{1cm} (5.14)

\[ \hat{D} = \sum_{i=1}^{N} \text{Var}(L_i). \] \hspace{1cm} (5.15)

Therefore, the optimal length for random splitting is equal to the mean of the length of the corresponding random segment. The minimum distortion is equal to the sum of the variance for each random segment. Practically, the optimal length for each segment should be an integer. Therefore, we round it up by: \[ \hat{k}_i = \lceil k_i \rceil \]
5.2.2.2  Optimal Key Points Selection

Given an arbitrary unknown query feature vector, the goal is to maximize the probability of matching key points in each segment of the query with the key points of the feature vector in the database that is similar to the query. Let us assume the beginning point \( w_i \) and the ending point \( w_{i+1} \) for the unknown segment is independent and identically distributed with the distribution \( P(w_i), P(w_{i+1}) \) and cumulative distribution function \( F \). The correct probability \( C \) is defined as the probability that all of three key points \( x_a, x_b, x_c \) are in the range \([w_i, w_{i+1}]\) for the unknown query segment.

\[
C = P(w_i \leq x_a \leq w_{i+1}) P(w_i \leq x_b \leq w_{i+1})
\]

\[
P(w_i \leq x_c \leq w_{i+1})
\]

\[
= P(w_i \leq x_a) P(w_{i+1} \geq x_a) P(w_i \leq x_b)
\]

\[
P(w_{i+1} \geq x_b) P(w_i \leq x_c) P(w_{i+1} \leq x_c)
\]

\[
= F(x_a)(1 - F(x_a)) F(x_b)(1 - F(x_b)) F(x_c)
\]

\[
(1 - F(x_c)) \tag{5.16}
\]

Since \( F(x_a)(1 - F(x_a)) \leq \left( \frac{F(x_a) + 1 - F(x_a)}{2} \right)^2 = \frac{1}{4} \) (the equality holds when \( F(x_a) = \frac{1}{2} \)) and similar properties can be applied to \( x_b, x_c \). Therefore, notice that the three optimal key points cannot be identical, maximizing the correct probability, we obtain:
Property 3: The optimal key points $x_a, x_b, x_c$ for the unknown segments should be chosen to make $|F(x_a) - \frac{1}{2}|, |F(x_b) - \frac{1}{2}|$ and $|F(x_c) - \frac{1}{2}|$ as small as possible. The sufficient condition is to choose three key points as adjacent points. If the distribution is symmetric, one of the optimal key points is the mid-points for random splitting.

5.3 Merging Procedures for LNS

Updating NSI efficiently corresponding to an existing Null Space based representation to accommodate updates while maintaining its affine view invariant nature is an important but challenging issue. In the traditional NS based representation such a merging is not possible unless raw data is available and the Null Space Invariants are recomputed by first combining all the raw data from the updates. For the proposed Localized Null Space Invariants such an operation is possible. In the following we present techniques to efficiently merge multiple Localized Null Spaces without requiring access to the raw data.

5.3.1 Exact Merging

Given two sub Null Spaces $Q_1, Q_2$ for non-overlapping feature matrices of size $L_1 \times (L_1 - 3)$ and $L_2 \times (L_2 - 3)$, respectively, when we merge these two subspaces, the new Null Space is of size $N \times (N - 3)$. As illustrated in Fig 3(b) by the shaded region, the missing information of nonzero elements corresponds to elements $q_j^{(i)}$ for $i = L_1, L_1 + 1, L_1 + 2$ and $j = 0, 1, 2$ in equation (2). We propose two approaches to recover the missing information. For simplicity, we discuss the case of merging two sub Null Spaces. However, the discussion can be easily extended to the case of merging $n$ sub Null Spaces.
5.3.1.1 Overlapped Segments

**Property 4:** When the merging subspaces have overlapped regions, and the length of the segments in each subspace is equal to or more than three, the corresponding Null Spaces can be merged perfectly. If the overlapping for sub Null Spaces is one or two, part of the missing information can be recovered.

**Merging Algorithm I:**

1. Given two overlapped sub null space invariant matrices $Q_1, Q_2$ of size $L_1 \times (L_1 - 3)$ and $L_2 \times (L_2 - 3)$, respectively, when $L_1 + L_2 \geq L + 3$, denote the elements in $Q_1, Q_2$ as $q_1^{(i)}$, $q_2^{(i)}$, respectively. If $L_1 + L_2 > L + 3$ and the null space invariant matrix to be merged is of the size $L \times (L - 3)$, then take the first $L + 3 - L_2$ rows and the first $L - L_2$ columns from $Q_1$ as $\tilde{Q}_1$.

2. Assign the null space invariants $q_j^{(i)} = q_1^{(i)}$ in $\tilde{Q}_1$ and $q_{j + L - L_2}^{(i)} = q_2^{(i)}$ and pad zeros to the rest of the positions in $Q$.

5.3.1.2 Reconstruction From Key Features Points

**Property 5:** Given non-overlapping Null Spaces, the necessary and sufficient condition for their perfect merging into one integrated Null Space is that the coordinates of three points in each feature matrix are given and at least one of the three points has been used as a *key point* to build the corresponding Null Space.

Without loss of generality, let us assume the coordinates of three known feature points are
\((u_0, v_0) = (c_1, c_2), (u_3, v_3) = (c_3, c_4), \) and \((u_4, v_4) = (c_5, c_6)\) where one of the key points is \((u_0, v_0)\). Thus, the coordinates of the other two key points can be computed:

\[
\begin{align*}
\hat{u}_1 &= \frac{(c_1 - c_5)(q_2^{(3)} + c_1c_4 - c_2c_3)}{(c_1 - c_5)(c_4 - c_2) - (c_1 - c_3)(c_6 - c_2)} - \frac{(c_1 - c_3)(q_2^{(4)} + c_1c_6 - c_2c_5)}{(c_1 - c_5)(c_4 - c_2) - (c_1 - c_3)(c_6 - c_2)}, \\
\hat{v}_1 &= \frac{(c_6 - c_2)(q_2^{(3)} + c_1c_4 - c_2c_3)}{(c_6 - c_2)(c_1 - c_3) - (c_1 - c_5)(c_4 - c_2)} - \frac{(c_6 - c_2)(q_2^{(4)} + c_1c_6 - c_2c_5)}{(c_6 - c_2)(c_1 - c_3) - (c_1 - c_5)(c_4 - c_2)}, \\
\hat{u}_2 &= \frac{(c_1 - c_5)(-q_1^{(3)} + c_1c_4 - c_2c_3)}{(c_1 - c_5)(c_4 - c_2) - (c_1 - c_3)(c_6 - c_2)} - \frac{(c_1 - c_3)(-q_1^{(4)} + c_1c_6 - c_2c_5)}{(c_1 - c_5)(c_4 - c_2) - (c_1 - c_3)(c_6 - c_2)}, \\
\hat{v}_2 &= \frac{(c_6 - c_2)(-q_1^{(3)} + c_1c_4 - c_2c_3)}{(c_6 - c_2)(c_1 - c_3) - (c_1 - c_5)(c_4 - c_2)} - \frac{(c_6 - c_2)(-q_1^{(4)} + c_1c_6 - c_2c_5)}{(c_6 - c_2)(c_1 - c_3) - (c_1 - c_5)(c_4 - c_2)}.
\end{align*}
\]

(5.17) \hspace{1cm} (5.18) \hspace{1cm} (5.19) \hspace{1cm} (5.20)

The equations \((4.17-4.20)\) indicate that the missing samples can be recovered from existing null space and a few key points, which could further be used as an important tool for merging multiple null space representations into an integrated null space representation.

**Merging Algorithm II:**

1. Compute the coordinates of the key points based on equations \((17-20)\) and recover the missing nonzero elements for Null Space Invariants in \(Q\).
2. Assign the null space invariants \( q_j^{(i)} = q_{1j}^{(i)} \) in \( Q_1 \) and \( q_{j, i+L-L_2}^{(i+L-L_2)} = q_{2j}^{(i)} \), and pad zeros to the rest of the positions in \( Q \).

5.3.2 **Approximate Merging**

Practically, it is possible that neither the subspaces are overlapped nor the key points are known. In this case, we propose an approximate solution for merging the null space invariants.

**Property 6:** Given \( N \) non-overlapped segments for Null Space a feature vector of the length \( L \), the merging process does not have enough information to compute \( \frac{3(N-1)}{L} \) information.

Based on Property 6, if \( L \gg N \), the impact of lost information is trivial and can be ignored for classification and retrieval when we merge the sub null spaces by padding zeros to the positions of lost information. The performance of approximate merging is shown in Section VIII.

5.4 **Perturbation Analysis**

Perturbation analysis is an important mathematical method which is widely used for analyzing the sensitivity of linear systems by adding a small term to the mathematical description of the exactly solvable problem. In 2d cases, let’s assume the noise matrix \( Z \) as:

\[
\begin{pmatrix}
\epsilon_{x,0} & \epsilon_{x,1} & \cdots & \epsilon_{x,N-1} \\
\epsilon_{y,0} & \epsilon_{y,1} & \cdots & \epsilon_{y,N-1} \\
0 & 0 & \cdots & 0
\end{pmatrix},
\]

where \( \epsilon_{x,i}, \epsilon_{y,i} \) have IID distribution with zero mean and the variance \( \delta^2 \). The sensitivity is measured by the ratio of the input error and the output error as (26) where the input error is referred to the error of the trajectory matrix and the output error is referred to the error of
the localized null space. We compute the error ratio for localized null space when the feature vector is split into 2 segments for 2d cases:

\[
\tau_L = \frac{1}{N}[K(r_{ab}^2 + r_{ac}^2 + r_{bc}^2) + \sum_{i=3}^{K+2} (r_{ai}^2 + r_{bi}^2 + r_{ci}^2)] \\
+ (N - K - 3)(r_{de}^2 + r_{df}^2 + r_{ef}^2 + \sum_{i=K+3}^{N-1} (r_{di}^2 + r_{ei}^2 + r_{fi}^2)),
\]

(5.22)

where \(a, b, c\) and \(d, e, f\) are the local key points selected for the first segment and the second segment respectively. The result for the error ratio can be easily extended into the case of \(N\) segments.

5.5 Performance Evaluation

5.5.1 Simulation Results for Splitting

Once we obtained the localized null space invariant representation for feature points provided by the splitting approach discussed in the earlier sections, we can rely on numerous methods for indexing and classification. We have selected a method for dimensionality-reduction and classification that is based on principal component null space analysis (PCNSA) (19). The benefits of the Localized Null Space Invariants are that they fully utilize the available information in each segments by choosing the local key points. In the traditional Null Space representation, the Null Space Invariants are computed relying on the same three key points. If in the partial query even one of the key points is not available, may be due to video occlusions, the classifi-
cation and retrieval completely fails.

We have applied the proposed approach to the classification and retrieval of motion trajectories when query trajectory has missing portions due to video occlusions. The data set used in our experiments has been obtained from Australian Sign Language (ASL) dataset (?) and Context Aware Vision using Image-based Active Recognition (CAVIAR) archives (?). We select 40 classes with 69 trajectories in each class from ASL dataset where the trajectories are obtained by registration of the hand coordinates at each successive instant of time by using a Power Glove interfaced to the system. Meanwhile, we choose 30,000 trajectories from the CAVIAR dataset where the trajectories represents different human actions, such as two people talking, shopping, chasing, etc. Figure 19 illustrates the top three retrieval results of motion trajectory corresponding to a shopping event when parts of the query trajectory are missing. Figure 20 demonstrates the top five retrieval results of motion trajectory corresponding to the query trajectory from the class "eat" which losts 10% samples. The retrieval results in Figure 19 and Figure 20 demonstrate the effectiveness of our LNS and associated splitting techniques for motion trajectory retrieval in the presence of affine transformations with partial query in large datasets. Specifically, the first three retrieval in Figure 19 are from the same type of human actions (shopping). Figure 20 indicates that the first three retrieval results for the query all come from the same class of sign language "eat".
5.5.2 Simulation Results for Merging

Next we evaluate the performance of our merging algorithm for non-overlapping segments as well as overlapping segments. The datasets used in this experiment have also been drawn from the ASL and CAVIAR database. Our algorithm can be applied to merge sub trajectories from the same orientation or from different orientations.

Figure 21 demonstrates the visual illustration in retrieval for merging three sub-trajectories in the motion event "walking" from CAVIAR dataset. We compare the precision and recall curves with overlapping and non-overlapping segments. As shown in Figure 22, the retrieval performance of the query consisting of overlapping segments is comparable to that of whole query trajectory. As expected, performance of the query trajectory consisting of non-overlapping segments with approximate merging is slightly worse than the query trajectory with overlapping segments. It has to be noted that merging segments with higher resolution gives lower dimensional space and therefore requires less computation time. We also evaluate the robustness of our approach in the experiment. Typically, the performance with different length of partial queries and in noisy case is shown in Figure 23. As can be shown in Figure 23, with the noise variance 0.2 and the query containing 80% samples, the proposed approach still works nicely. The proposed framework can also be utilized for merging databases instead of queries. With respect to retrieval time, we compare the performance of CPU time using localized null space and traditional null space for merging process with different number of segments in Figure 24. Since the traditional null space requires to compute the new null space from scratch, it consumes much more computational time than the proposed Localized null space.
Figure 19. Top three retrieval results for a partial query trajectory containing only 75% samples with the missing portion from CAVIAR dataset.
Figure 20. Top five retrievals corresponding to the partial query for motion trajectory from ASL dataset which losts 10% samples.
Figure 21. Top three retrieval results for merging three sub-trajectories as the query containing only 60% samples.

Figure 22. The comparison of precision and recall curves in retrieval with curvature scale space (CSS) and centroid distance function (CDF) based approaches.
Figure 23. The comparison of average precisions for retrieval of different types of motion events in CAVIAR dataset with 20% missing data.

Figure 24. The comparison of cpu time for merging process with different number of segments using ASL dataset.
CHAPTER 6

NON-LINEAR KERNEL SPACE INVARIANTS

Among view-invariant systems, majority of them represent affine view-invariance in a linear space, thus limiting their applicability to data that may have only linear transformations, e.g. weak perspective transformation model. However, weak perspective model is only suitable and valid for data capturing environments that have small focal length, small field of view, and small depth variation. If the depth of an object has large variations, representing the feature points of the object with the average depth, assumed in the weak perspective model, will lose significant information and cause inaccurate results for classification and retrieval. Moreover, if the x and y coordinates are not small compared to the depth, namely the field of view is not small enough, weak perspective model is also invalid for mapping 3D data onto 2D images. In this case, the weak perspective transformation does not hold and the standard perspective transformation is required. The standard perspective model will result in nonlinear mapping of the 3D environment onto 2D images.

We propose a novel fundamental mathematical framework for non-linear invariant representation and use this framework for the important application of view-invariant classification and retrieval of motion events involving multiple motion trajectories. We mainly discuss the use of Taylor expansion for determining NKSI. The approach can be easily extended to other expansion methods (e.g., Chebyshev polynomial and Lagrange polynomial). We first provide a theoretical analysis of the basic invariance property of the proposed non-linear in-
variant space representation. Let us assume we have a non-linear transformation
\[ f(x_i, y_i) = p_1 \times f_1(x_i, y_i) + p_2 \times f_2(x_i, y_i) + \ldots + p_t \times f_t(x_i, y_i) + \ldots + p_k \times f_k(x_i, y_i), \]
where the coefficients \( p_t \) are random variables and \( f_t(x_i, y_i) \) are non-linear transformations of the raw data \( x_i, y_i \) with deterministic forms. For example, in the case of \( k = 2 \), the original raw data matrix \( M \) with \( n \) samples is transformed into a new raw data matrix \( \tilde{M} \) with the dimension \( 3 \times n \) as
\[
\tilde{M} = \begin{pmatrix}
  f_1(x_0, y_0) & \ldots & f_1(x_{n-1}, y_{n-1}) \\
  f_2(x_0, y_0) & \ldots & f_2(x_{n-1}, y_{n-1}) \\
  1 & 1 & 1
\end{pmatrix}
\]
Once we have obtained the new raw data \( \tilde{M} \), the non-linear kernel space invariants (NKSI) can be computed as:
\[
q_0^i = \det \begin{pmatrix}
  f_1(x_1, y_1) & f_1(x_2, y_2) & f_1(x_i, y_i) \\
  f_2(x_1, y_1) & f_2(x_2, y_2) & f_2(x_i, y_i) \\
  1 & 1 & 1
\end{pmatrix}
\] (6.1)
\[
q_1^i = -\det \begin{pmatrix}
  f_1(x_0, y_0) & f_1(x_2, y_2) & f_1(x_i, y_i) \\
  f_2(x_0, y_0) & f_2(x_2, y_2) & f_2(x_i, y_i) \\
  1 & 1 & 1
\end{pmatrix}
\] (6.2)
\[ q_i^k = \det \begin{pmatrix} f_1(x_0, y_0) & f_1(x_1, y_1) & f_1(x_i, y_i) \\ f_2(x_0, y_0) & f_2(x_1, y_1) & f_2(x_i, y_i) \\ 1 & 1 & 1 \end{pmatrix} \] (6.3)

\[ q_i^j = -\det \begin{pmatrix} f_1(x_0, y_0) & f_1(x_1, y_1) & f_1(x_2, y_2) \\ f_2(x_0, y_0) & f_2(x_1, y_1) & f_2(x_2, y_2) \\ 1 & 1 & 1 \end{pmatrix}, \] (6.4)

\[ q_i^j = 0 \text{ for } j = 3, 4, \ldots, i-1, i+1, \ldots, n-1 \] (6.5)

Moreover, it is easy to extend the mathematical formulation of (3) for the case of 3D points.

As it is known, the standard perspective object-to-image transformation \( \pi_{A,\xi} \) has the form

\[
\begin{pmatrix} u' \\ v' \end{pmatrix} = \pi'_{A,\xi} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \frac{\lambda}{(gx + hy + kz + \xi_3 + \lambda)} 
\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \xi] \] (6.6)
where $\lambda$ is the focal length and the transformation

$$A = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & k \end{pmatrix}$$

(6.7)

Therefore, we have:

$$u'_i = \lambda (ax_i + by_i + cz_i + \xi_1) \quad (6.8)$$

$$v'_i = \lambda (dx_i + ey_i + fz_i + \xi_2) \quad (6.9)$$

where $i = 0, ..., N - 1$.

### 6.0.3 Basis Decomposition

From the equations (8) and (9), the standard perspective transformation is a non-linear transformation. Generally, we can formulate the expansions as:

$$u'_i \approx \sum_{p=0}^{N} p_{u,k} \Phi_{u,k}(x_i, y_i), \quad v'_i \approx \sum_{p=0}^{N} p_{v,k} \Phi_{v,k}(x_i, y_i), \quad (6.10)$$

where $\Phi_{u,k}(x_i, y_i), \Phi_{v,k}(x_i, y_i)$ is a basis which is a function of $x_i$ and $y_i$, $p_{u,k}, p_{v,k}$ are the corresponding coefficients for the basis. It is well known that Karhunen-Loeve Transform (KLT) \(\text{(?)}\) gives the optimal expansions where $\Phi_k(x_i, y_i)$ here represents the eigenfunction of the transformation $f_u(x_i, y_i)$, $f_v(x_i, y_i)$. The computation of optimal $\Phi_{u,k}(x_i, y_i), \Phi_{v,k}(x_i, y_i)$ requires that the covariance
matrix of the transformation function is known. If covariance matrix of the non-linear function is not available, we consider the widely used expansion, such as Taylor expansion at the original point:

\[
\begin{align*}
\mathbf{u}'_i & \approx \sum_{w=0}^{N} q_{u,k} x_i^m y_i^n, \\
\mathbf{v}'_i & \approx \sum_{w=0}^{N} q_{v,k} x_i^m y_i^n,
\end{align*}
\]  

(6.11)

where \(m + n = w\). It has to be noted that we could always normalize the coordinates of all the trajectories to the range \([0, 1]\), therefore the high order terms \(q_{u,k} x_i^m y_i^n, q_{v,k} x_i^m y_i^n\), where \(m + n > N\), is small enough and can be ignored. In this case, the transformed raw data matrix becomes:

\[
\tilde{\mathbf{M}} = \begin{pmatrix}
    x_0 & \ldots & x_{n-1} \\
    y_0 & \ldots & y_{n-1} \\
    x_0^2 & \ldots & x_{n-1}^2 \\
    y_0^2 & \ldots & y_{n-1}^2 \\
    x_0 y_0 & \ldots & x_{n-1} y_{n-1} \\
    \ldots & \ldots & \ldots \\
    1 & 1 & 1
\end{pmatrix}
\]

Since we have obtained the transformed raw data matrix \(\tilde{\mathbf{M}}\), we can compute the NKSI based on \(\tilde{\mathbf{M}}\). Besides Taylor expansion, we also test the performance of NKSI in classification and
retrieval with Chebyshev polynomial and Lagrange polynomial. Specifically, the Chebyshev polynomial of the first kind can be represented as:

\[ T_0(x) = 1, \ T_1(x) = x, \ T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) \]  \hspace{1cm} (6.12)

Once we have obtained the NKSI, there are various classification and retrieval algorithms that can be applied on NKSI. We choose a highly discriminative method, referred to as principal component null space analysis (PCNSA), described in (19).
7.1 Introduction

The advent of multimedia technology and popular online multimedia archives in recent years have raised many interesting research questions in video event/object classification and retrieval (26). However, several fundamental challenges in video classification and retrieval systems must be addressed before such systems can be employed in critical applications. Firstly, one of the main challenges is that events/objects captured from different viewpoints or moving camera sensors lead to completely different representations. Secondly, due to the multi-modality of multi-camera system, the dimension in each raw data matrix is different, e.g., the length of each trajectory is significantly different as shown in Figure 25 where the length of trajectory A is much longer than the length of the trajectory B. The traditional way of normalizing trajectories by either resampling the trajectories and Fourier transformation based methods of trajectories introduce the significant loss of information (26). Another example is that for 3D object retrieval, generally the number of extracted feature points from different objects are not identical. Taking the minimum number of SIFT feature points in order to formulate the raw data matrix introduce unrecoverable loss of accuracy for information retrieval. Third, for multiple trajectory retrieval, it is often the case that the number of trajectories is greater than the length of each trajectory. For instance, as shown in Figure 26, we have 18 trajectories in
Figure 25. Unequal multiple trajectory representation: A video frame and its rotated version containing three trajectories with significantly different length, where trajectories A, B, and C have 45, 12, and 26 samples, respectively.

one frame for the motion event in the crowd and the maximum length of the trajectory is 15. In this case, the null space invariant representation for \( AX = 0 \) where \( A \) is the raw data matrix and \( X \) is the null space does not have the non-zero solution, because the number of columns in matrix \( A \) is larger than the number of rows, therefore, \( AX = 0 \) is an over-determined equation.

Specifically, we propose a new Bilinear Invariants (BI) representation and present its enormous potential in multimedia information retrieval. The resulting representation could be applied to invariant multimedia information retrieval with transformations from different dimensions simultaneously and the database with multi-modality in a multi-camera system.
7.2 Invariants Theory

7.2.1 T-Invariants

In functional analysis, the kernel \( \text{Ker}(\Psi) \) of a general operator \( \Psi : V \to W \) is the set of all operands \( v \in V \) such that \( \Psi(v) = 0 \); i.e.

\[
\text{Ker}(\Psi) = \{ v \in V : \Psi(v) = 0 \}
\]

where we use \( 0_W \) to denote the null vector in \( W \). Let us consider a transformation \( T : W \to Z \) and use \( 0_Z \) to denote the null vector in \( Z \). We say that the kernel \( \text{Ker}(\Psi) \) is \( T \)-invariant if \( T \) is a bijection and \( 0_W \) is in the kernel of the transformation \( T \); i.e. \( T(0_W) = 0_Z \). Note that in this case the kernel \( \text{Ker}(T \circ \Psi) \) of the transformed operator \( \Psi \) is identical to the original
kernel $\text{Ker}(\Psi)$; i.e. $\text{Ker}(T \circ \Psi) = \{v \in V : T \circ \Psi(v) = 0_Z\} = \{v \in V : \Psi(v) = T^{-1}(0_Z)\} = \{v \in V : \Psi(v) = 0_W\} = \text{Ker}(\Psi)$.

### 7.2.2 Linear Invariants

A special case of the $T$ – invariants theory introduced above has been explored for linear operators (26). We note that the linear invariants theory extends to affine transforms in homogeneous coordinate system (26). Let us consider a linear Operator $L : \mathbb{R}^n \rightarrow \mathbb{R}^m$. In this case, the kernel space is referred to as null space $N(L)$ of the linear operator $L$; i.e.

$$N(L) = \{x \in \mathbb{R}^n : L(x) = 0_m\},$$

where $0_m$ is zero in the vector space $\mathbb{R}^m$. We note that the null space $N(L)$ is invariant to an invertible linear transformation $M : \mathbb{R}^m \rightarrow \mathbb{R}^l$; i.e.

$$N(ML) = \{x \in \mathbb{R}^n : ML(x) = 0_l\} = \{x \in \mathbb{R}^n : L(x) = M^{-1}(0_l)\} = \{x \in \mathbb{R}^n : L(x) = 0_m\} = N(L).$$

Furthermore, an efficient representation of a basis of the null space has been presented in (26).

### 7.2.3 Bilinear Invariants

In the following, we will explore an important special case of invariant theory for bilinear transformations and determine an efficient basis representation of its kernel space. Let us consider a bilinear transformation $L$ given by

$$L(V) = AVB,$$  \hspace{1cm} (7.2)

where $A$ and $B$ are $m \times n$ and $p \times q$ raw data matrices, respectively, and $V$ is an $n \times p$ matrix.

The kernel $\text{Ker}(L)$ is given by

$$\text{Ker}(L) = \{V \in \mathbb{R}^{n \times p} : L(V) = AVB = 0_{m \times q}\},$$  \hspace{1cm} (7.3)
where $0_{m\times q}$ is an $m \times q$ matrix containing the values 0.

Kernel $\text{Ker}(L)$ can be equivalently characterized by

$$\text{Ker}(L) = \{ V \in \mathbb{R}^{n \times p} : (B^T \otimes A)\text{vec}(v) = 0_{m \times q} \}, \quad (7.4)$$

where $\otimes$ is used to denote the Kronecker product and $\text{vec}(v) \in \mathbb{R}^{np}$ is the vectorized matrix $V$. Let us use $M$ to represent the matrix given by

$$M = (B^T \otimes A) = \begin{pmatrix}
m_{11} & m_{12} & \cdots & m_{1t} \\
m_{21} & \cdots & \cdots & m_{2t} \\
\vdots & \cdots & \cdots & \cdots \\
m_{s1} & m_{s2} & \cdots & m_{st} 
\end{pmatrix} \quad (7.5)$$

We now represent bilinear invariants (BI) of the kernel space $\text{Ker}(L)$ of the bilinear transformation $L$ as the basis of the null space $N(M)$ of the composite linear matrix $M$ (26). Specifically, the bilinear invariants basis is characterized by the collection of matrices $V_i$ whose vectorized form $v^{(i)}$ is given by

$$v^{(i)}_1 = \det \begin{pmatrix}
m_{12} & m_{13} & \cdots & m_{1s} & m_{1,i} \\
m_{22} & m_{23} & \cdots & m_{2s} & m_{2,i} \\
\vdots & \cdots & \cdots & \cdots & \cdots \\
m_{s2} & m_{s3} & \cdots & m_{ss} & m_{s,i} 
\end{pmatrix},$$
\[ v_2^{(i)} = -\det \begin{pmatrix} m_{11} & m_{13} & \cdots & m_{1s} & m_{1,i} \\ m_{21} & m_{23} & \cdots & m_{2s} & m_{2,i} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ m_{s1} & m_{s3} & \cdots & m_{ss} & m_{s,i} \end{pmatrix}, \]

\[ v_s^{(i)} = (-1)^s \det \begin{pmatrix} m_{11} & m_{12} & \cdots & m_{1,s-1} & m_{1,i} \\ m_{21} & m_{22} & \cdots & m_{2,s-1} & m_{2,i} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ m_{s1} & m_{s2} & \cdots & m_{s,s-1} & m_{3,i} \end{pmatrix}, \]

\[ v_k^{(i)} = 0, \forall k = s + 1, \ldots, t. \]

**Lemma 1:** Given a left- and right-multiply linear transformations \( T_A \) and \( T_B \) on the matrices \( A \) and \( B \), respectively. The bilinear invariant basis is simultaneously invariant to the left- and right-multiply linear transformations \( T_A \) and \( T_B \).

**Proof:** When we apply the linear transformations \( T_A \) and \( T_B \) on \( A \) and \( B \) simultaneously, we have

\[(T_A A) V(B T_B) = ((B T_B)^T \otimes (T_A A)) \text{vec}(v) =
\]

\[(T_B^T B) \otimes (T_A A) \text{vec}(v) =
\]

\[(T_B^T \otimes T_A)(B^T \otimes A) \text{vec}(v) = 0, \quad (7.6)\]

where we have used the mixed-product property for Kronecker product. ■
7.3 Tensor Multilinear Invariants

We shall now extend bilinear invariants to tensors in the event that there are more than two raw data matrices which are subject to distinct linear transformations along different dimensions. Let us consider $N$ raw data matrices $A_1, A_2, \ldots, A_N$, where the data matrix $A_i$ has the dimension $a_i \times b_i$, for $1 \leq i \leq N$. The tensor multilinear transformation $L$ is given by

$$L(X) = X \times_1 A_1 \times_2 A_2 \times \cdots \times_n A_n,$$

(7.7)

where $\times_i$ is used to denote the $i$th-mode product (2). Kernel $\text{Ker}(L)$ of the tensor multilinear transformation $L$ is given by

$$\text{Ker}(L) = \{ X \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_N} : L(X) = X \times_1 A_1 \times_2 A_2 \times \cdots \times_n A_n = 0 \}.$$  

(7.8)

The kernel of the tensor multilinear invariants are as follows:

$$(A_n^T \otimes A_{n-1}^T \otimes \cdots \otimes A_1)\text{vec}(X) = 0.$$  

(7.9)

Similar to our previous approach for the representation of bilinear invariants, we proceed to represent the basis of the kernel space $\text{Ker}(L)$ of the tensor bilinear transformation $L$ as the basis of the null space of the linear operator $(A_n^T \otimes A_{n-1}^T \otimes \cdots \otimes A_1)$ (26).
7.4 Reduced-Dimension Multilinear Invariants

In Sections 2 and 3, we presented a method for matrix and tensor representation of multilinear invariants that provides the complete invariant basis for bilinear transformations. This approach is essential for characterization of invariant representations when visual data such as motion trajectories have different lengths or feature dimension. Nonetheless, the computational complexity of the proposed multilinear invariants is high and could be prohibitive in video retrieval and classification applications. Here, we propose a novel solution to reduced-dimension multilinear invariant representation. We demonstrate experimentally that reduced-dimension bilinear invariants can achieve high-accuracy and low computational complexity in multimedia information retrieval and classification. We formulate reduced-dimension multilinear invariants by considering the bilinear transformation \( L \) given by (2) such that the matrix \( V \) has rank \( w \); i.e. the kernel \( \text{Ker}(L) \) is given by

\[
\text{Ker}(L) = \{ V \in \mathbb{R}^{n \times p} : L(V) = AVB = 0, \text{rank}(V) = w \},
\]

(7.10)

where \( A, B, \) and \( V \) are \( m \times n, p \times q \), and \( n \times p \) matrices, respectively, and \( w < \min(n, p) \).

The solution to (10) is equivalent to the solution of the equation \( AXYB = 0 \), where \( A, X, Y, B \) are \( m \times n, n \times w, w \times p, \) and \( p \times q \) matrices, respectively. We derive this solution by first determining the traditional null-space invariants of \( A \) by solving \( AX = 0 \) to obtain \( m - n \) basis \( X = [x_1, x_2, \ldots, x_{m-n}] \). We then characterize the null-space invariants of \( B^T \) by solving
$\mathbf{B}^\top \mathbf{Y}^\top = 0$ to obtain $p - q$ basis $\mathbf{Y} = [y_1, y_2, \ldots, y_{p-q}]$. We finally form the rank-constraint invariant space $\mathbf{V}$ given by $\mathbf{V} = [v_1, v_2, \ldots, v_w]$, where $v_i = x_i y_i^\top$. 
CHAPTER 8

CONCLUSION AND FUTURE WORK

In this dissertation, we have developed a novel robust classification and retrieval system for motion trajectories in video sequences relying on null space invariants. We demonstrated the enormous potential of the NSI operator as a powerful view-invariant representation for recognition and retrieval in the presence of arbitrary moving cameras. The computational complexity of NSI is very low and it possesses several important features, e.g. generality, robustness, and preserves information of the original data when computing invariants. We considered the scenarios of fixed camera with unknown views and arbitrary moving cameras. For noisy data sets, we derived the perturbed null operators with perturbation theory and analyzed the sensitivity of the systems in terms of error ratio and SNR. Given the perturbation analysis, optimal sampling schemes are proposed to guarantee the robustness of the system. Moreover, null space representation of segmented trajectories are proposed for moving cameras. Typically, overlapping segmentation and non-overlapping segmentation approaches for classification and retrieval are investigated when the motion events are taken from arbitrary moving cameras. The proposed approach could also be applied for object recognition. Moreover, we compare our approaches with traditional classification and retrieval approaches. Simulation results demonstrate the effectiveness and superiority of our approaches. We further propose a novel framework of tensor null space for classification and retrieval of high order data.

Moreover, we have considered the splitting and merging of null space view invariant represen-
tation in the dynamical video database. Namely, the view invariant classification and retrieval with partial queries and dynamical updating. We present a novel robust multi-dimensional Localized Null Space and associated dynamic updating and downdating techniques, thus allowing classification and retrieval in the presence of affine transformations and partial information.

We investigated the robustness of Localized Null Space (LNS) using perturbation analysis. We further determined the optimal segmentation of the data by minimizing a distortion criterion. We demonstrated the effectiveness and robustness of the proposed techniques for motion event classification and retrieval applications by posing different affine transformations of partial queries.

Finally, we have proposed the Non-linear Kernel Space Invariants (NKSI) for non-linear transformation of the raw data and Multilinear Invariants for view invariant retrieval of raw data with unequal length of different dimensions. We also extended the concept of Multilinear Invariants to Tensor Multi-linear Invariants for high dimensional data. We provided the simulation results to demonstrate the effectiveness of our approach.

In the future work, we plan to find additional interesting applications of null space invariants in other areas, for example, the invariants for partial differential equations. Another possible direction is to incorporate semantic information into our framework to improve the retrieval performance. Moreover, we are also interested in applying null space invariants on more visual image and video features as well as other texture features to see whether or not they can improve the performance of classification and retrieval.
CHAPTER 9

REFERENCES


7. E. Keogh, J. Lin, and A. Fu, HOT SAX: Efficiently finding the most unusual time series subsequence, in Proc. 5th IEEE Int. Conf. Data Mining (ICDM 2005), Houston, TX, Nov. 27C30, 2005.


CHAPTER 10

APPENDICES

10.1 The proof for the convergence of the mean of the error ratio with Poisson sampling

Due to the tedious derivations, here we show the main steps of the proofs. Since we know that with Poisson sampling $\lambda = N$, $E(t^n_k) = \frac{(n+k-1)!}{N^{n+1}(k-1)!}$, where $N$ is the total number of samples and $n$ is the index for Taylor expansion. So we have:

$$
\frac{1}{N} \sum_{i=3}^{N} E(t^n_k) = \frac{1}{N} \frac{(k+2)!}{2!} + \frac{(k+3)!}{3!} + \ldots + \frac{(k+N-1)!}{(N-1)!} \quad (10.1)
$$

Denoting $C_m^n = \frac{m!}{n!(m-n)!}$, we have:

$$
\frac{1}{N} \sum_{i=3}^{N} E(t^n_k) = \frac{C_{k+2}^2 + C_{k+3}^3 + \ldots + C_{k+N-1}^{N-1}}{N^{k+1}/k!} \quad (10.2)
$$

According to the basic property $C_m^n + C_m^{n-1} = C_m^{n+1}$, it can be further simplified:

$$
\frac{1}{N} \sum_{i=3}^{N} E(t^n_k) = \frac{C_{k+2}^1 + C_{k+2}^2 + \ldots + C_{k+N-1}^{N-1} - C_{k+2}^1}{N^{k+1}/k!} = \frac{C_{k+2}^2 + C_{k+3}^3 + \ldots + C_{k+N-1}^{N-1} - C_{k+2}^{N-1}}{N^{k+1}/k!} = \frac{C_{k+N}^{N-1} - C_{k+2}^1}{N^{k+1}/k!} \quad (10.3)
$$
Therefore,

\[
\lim_{N \to \infty} \frac{1}{N} \sum_{i=3}^{N} E(t_i^k) = \lim_{N \to \infty} \frac{C_{N-1}^{N-k-1} - C_{k+1}^{N+1}}{N^{k+1}/k!} = \lim_{N \to \infty} \frac{C_{k+N}^{N-1}}{N^{k+1}/k!} = \lim_{N \to \infty} \frac{(k + N)!k!}{(k + 1)!(N - 1)!}.
\]

\[ \frac{1}{k + 1} \quad (10.4) \]

\[
\sum_{j=3}^{N-1} E(r_j^2) = \sum_{j=3}^{N-1} E\left\{ f'(0)t_j + \ldots + \frac{f^{(n)}(0)}{n!}t_j^n \right\}^2 + \sum_{j=3}^{N-1} E\left\{ g'(0)t_j + \ldots + \frac{g^{(n)}(0)}{n!}t_j^n \right\}^2.
\]

\[ \sum_{j=3}^{N-1} E(r_j^2) = \sum_{j=3}^{N-1} \sum_{k=1}^{2n} \frac{(a_k + b_k)T^k}{k + 1}, \quad (10.5) \]

we obtain

\[
\lim_{N \to \infty} \frac{1}{N} \sum_{j=3}^{N-1} E(r_j^2) = \sum_{k=1}^{2n} \frac{(a_k + b_k)T^k}{k + 1}, \quad (10.6)
\]

where \(k\) is the index for Taylor series and for \(a_k\) and \(b_k\), if \(k\) is odd,

\[
a_k = \sum_{i=1}^{k-1} \frac{2g^{(i)}(0)g^{(k-i)}(0)}{i!(k-i)!}, \quad b_k = \sum_{i=1}^{k-1} \frac{2f^{(i)}(0)f^{(k-i)}(0)}{i!(k-i)!}, \quad (10.7)
\]

if \(k\) is even,

\[
a_k = \sum_{i=1}^{k-1} \frac{2g^{(i)}(0)g^{(k-i)}(0)}{i!(k-i)!} + \left( \frac{f^{(i)}}{i!} \right)^2, \quad b_k = \sum_{i=1}^{k-1} \frac{2f^{(i)}(0)f^{(k-i)}(0)}{i!(k-i)!} + \left( \frac{g^{(i)}}{i!} \right)^2. \quad (10.8)
\]
Similarly, we can obtain

\[
\lim_{N \to \infty} \frac{1}{N} \sum_{j=3}^{N-1} E(r_j^2) = \lim_{N \to \infty} \frac{1}{N} \sum_{j=3}^{N-1} E(r_j^2) = \lim_{N \to \infty} \frac{1}{N} \sum_{j=3}^{N-1} E(r_{j_2}^2),
\]

(10.9)

Therefore, it is easy to show that:

\[
\lim_{N \to \infty} \frac{1}{N} (N-3) E(r_{01}^2 + r_{02}^2 + r_{12}^2) = 0.
\]

(10.10)

Q.E.D.

10.2 The proof for the convergence of SNR with uniform sampling

It is easy to show that with uniform sampling, both of the nominator and denominator are \(O(N^{2n+1})\). Thus, the limit of the SNR only depends on the coefficients of \(N^{2n+1}\):

\[
\lim_{N \to \infty} \text{SNR} = \lim_{N \to \infty} \frac{T^{2n} (A(\frac{g^{(n)}(0)}{n!})^2 + B(\frac{f^{(n)}(0)}{n!})^2 + C\frac{f^{(n)}(0)g^{(n)}(0)}{n!} \sum_{i=3}^{N-1} i^{2n})}{6\delta^2 T^{2n} [\frac{g^{(n)}(0)}{n!}]^2 + \frac{f^{(n)}(0)}{n!}^2] \sum_{i=3}^{N-1} i^{2n}}
\]

(10.11)

Q.E.D.

10.3 The proof for the convergence of the variance of the error ratio with Poisson sampling

\[
\text{Var}(t^2_k) = E(t^2_k) - [E(t^2_k)]^2 = \frac{(2n+k-1)!(k-1)! - [(n+k-1)]^2}{N^{2n}[(k-1)!]^2}
\]

(10.12)
We first prove that

\[
\lim_{N \to \infty} \frac{1}{N} \sum_{k=3}^{N} \text{Var}(t_k^n) = 0 \tag{10.13}
\]

Since we know that

\[
\lim_{N \to \infty} \frac{1}{N} \sum_{k=3}^{N} \text{Var}(t_k^n) \leq \lim_{N \to \infty} \frac{(2n + 2)^{2n} + \ldots + (2n + N - 1)^{2n} - (3^{2n} + \ldots + N^{2n})}{N^{2n+1}} \tag{10.14}
\]

Let us denote \( T(k) = (2n + k - 1)^{2n} - k^{2n} \),

\[
T(k + 1) - T(k) = [(2n + k)^{2n} - (2n + k - 1)^{2n}] - [(k + 1)^{2n} - k^{2n}]
\]

\[
= [(2n + k)^{2n-1} + (2n + k)^{2n-2}(2n + k - 1) + \ldots + (2n + k - 1)^{2n-1}] -
\]

\[
[(k + 1)^{2n-1} + (k + 1)^{2n-2}k + \ldots + k^{2n-1}] > 0 \tag{10.15}
\]

Namely, \( T(k) \) monotonically increases with \( k \). Thus,

\[
\lim_{N \to \infty} \frac{(2n + 2)^{2n} + \ldots + (2n + N - 1)^{2n} - (3^{2n} + \ldots + N^{2n})}{N^{2n+1}} \leq
\]

\[
\lim_{N \to \infty} \frac{(N - 2)[(2n + N - 1)^{2n} - N^{2n}]}{N^{2n+1}} = 0 \tag{10.16}
\]

Since \( \lim_{N \to \infty} \frac{1}{N} \sum_{k=3}^{N} \text{Var}(t_k^n) \geq 0 \), so

\[
\lim_{N \to \infty} \frac{1}{N} \sum_{k=3}^{N} \text{Var}(t_k^n) = 0 . \tag{10.17}
\]
Furthermore, we have:

$$\text{Var}(t_{m}^{n}t_{k}^{n}) = E(t_{j}^{2m}t_{k}^{2n}) - [E(t_{j}^{m}t_{k}^{n})]^{2} = E(t_{j}^{2m})E(t_{k}^{2n}) - [E(t_{j}^{m})E(t_{k}^{n})]^{2}$$

$$= \frac{(2m + j - 1)!(2n + k - 1)!}{N^{2m}N^{2n}(j - 1)!(k - 1)!} - \left(\frac{(m + j - 1)!(n + k - 1)!}{N^{m+n}(j - 1)!(k - 1)!}\right)^{2}$$

(10.18)

According to the equation (17), \(j = 0, 1, 2, k = 3, ..., N - 1\). For \(j = 1\),

$$\lim_{N \to \infty} \frac{1}{N} \sum_{k=3}^{N} C_{k} \text{Var}(t_{j}^{m}t_{k}^{n}) < \lim_{N \to \infty} \sum_{k=3}^{N} C_{k} \frac{2m!(2n + k - 1)^{2n} - (m!)^{2}2^{2n}}{N^{2(m+n)+1}} <$$

$$\frac{N - 3}{N} \max(C_{k}) \frac{2m!(2n + N - 1)^{2n}}{N^{2(m+n)}} = 0 ,$$

(10.19)

where \(C_{k}\) is the coefficient of \(\text{Var}(t_{j}^{m}t_{k}^{n})\). Similarly, it is easy for \(j = 0, 2\),

$$\lim_{N \to \infty} \frac{1}{N} \sum_{k=3}^{N} C_{k} \text{Var}(t_{j}^{m}t_{k}^{n}) = 0$$

(10.20)

Since we know that

$$\text{Var}\left(\sum_{i=1}^{n} X_{i}\right) = \sum_{i=1}^{n} \text{Var}(X_{i}) + 2 \sum_{i,j} \text{Cov}(X_{i}, X_{j})$$

(10.21)

For \(\sum_{j=3}^{N-1} \text{Cov}(t_{j}^{p}t_{m}^{q}, t_{j}^{p}t_{m}^{l})\), when \(m = 1\),

$$\sum_{j=3}^{N-1} \text{Cov}(t_{j}^{p}t_{m}^{q}, t_{j}^{p}t_{m}^{l}) = \sum_{j=3}^{N-1} (E(t_{j}^{p+s})E(t_{j}^{q+t}) - E(t_{j}^{p})E(t_{j}^{q})E(t_{j}^{l}))$$
\[ \sum_{j=3}^{N-1} \left( \frac{(p + s + j - 1)! (q + 1)!}{N^{p+s+q+1}(j-1)!} - \frac{(p + j - 1)! (s + j - 1)! q!!}{N^{p+s+q+1}(j-1)!^2} \right) \]

Since the nominator is \( O(N^{p+s}) \) and the denominator is \( O(N^{p+s+q+1}) \), so we have

\[ \lim_{N \to \infty} \sum_{j=3}^{N-1} \text{Cov}(t_p^j t_q^j, t_s^j t_l^j) = 0 \]

(10.23)

Based on that, it is easy to know that

\[ \lim_{N \to \infty} \text{Var}(\tau) = 0 \]

(10.24)

Q.E.D.
CHAPTER 11

VITA
Xu Chen
University of Illinois at Chicago, Chicago, IL
Electrical and Computer Engineering

Email: xchen27@uic.edu
Website: http://ece.uic.edu/~xchen/

Ph.D University of Illinois at Chicago (UIC), 08/2006 – present
Department of Electrical and Computer Engineering (ECE),
Multimedia Communications Laboratory (MCL) and Multimedia Systems Laboratory (MSL),
GPA 3.7/4.0 United States

Department of Electrical Engineering (EE), GPA 3.8/4.0 China

Research Assistant with Prof. Dan Schonfeld and Prof. Ashfaq Khokhar (co-advisor) 08/2006 – 05/2010
Department of Electrical and Computer Engineering, University of Illinois at Chicago, Chicago

Journal Papers


Conference Papers

1 Xu Chen, Dan Schonfeld, and Ashfaq Khokhar, BILINEAR INVARIANT REPRESENTATION FOR MULTIMEDIA INFORMATION CLASSIFICATION AND RETRIEVAL”, International Conference on Image Processing (ICIP2010), Hong Kong, 2010.


3 Xu Chen, Dan Schonfeld, and Ashfaq Khokhar, ”Localized Null-Space Representation For Dynamic Updating and Downdating In Image and Video Databases”, International Conference on Image Processing (ICIP’2009), Cairo, Egypt, 2009.

4 Xu Chen, Dan Schonfeld, and Ashfaq Khokhar, "View-Invariant Tensor Null-Space Representation For Multiple Motion Trajectory Retrieval and Classification”, International Conference on Acoustics, Speech and Signal Processing (ICASSP’2009), Taipei, Taiwan, 2009.

5 Xu Chen, Dan Schonfeld, and Ashfaq Khokhar, "Robust Closed-form Localization of Mobile Targets Using a Single Sensor Based on a Non-linear Measurement Model”, 9th IEEE
Workshop on Signal Processing Advances in Wireless Communications, (SPAWC’08), Recife, Brazil, 2008.


- IEEE Student Member, 2007-2008, WHO’S WHO IN AMERICA in Oct,2009
- UIC travel award and Graduate Student Council (GSC) award for 2008, Student Travel Grant for IEEE 2007 Statistical Signal Processing Workshop
- Reviewer for ICC, GLOBECOM and ICME, Outstanding Student Award of SJTU, 2003-05
- Windows, Linux, Solaris, C/C++, Assembly, Matlab, Java, Qt, SQL, HTML, OpenCV, Mathematica, Mathtype, ElectronicWorkbench, Dreamweaver, MAC protocol, CAD
- Video and Image Processing, Computer Vision Video Tracking, Pattern Recognition, Machine Learning, Video and Image Coding, Sensor network, Data Analysis