BILINEAR INVARIANT REPRESENTATION FOR MULTIMEDIA INFORMATION CLASSIFICATION AND RETRIEVAL

Xu Chen, Dan Schonfeld and Ashfaq Khokhar

Department of Electrical and Computer Engineering, University of Illinois at Chicago
{xchen27, dans, ashfaq}@uic.edu

ABSTRACT
In this paper, we present a novel bilinear invariant representation for multimedia information classification and retrieval. We introduce the concept of kernel space from functional analysis to show that null space invariants are only the special case when the transformation is linear. Subsequently, we derive the exact invariant basis for one of the important applications for kernel space: bilinear invariants. We demonstrate that the proposed bilinear invariant basis provides a more powerful tool than the previous null space invariants in multimedia information retrieval and classification. The simulation results demonstrate that the proposed bilinear invariant basis provides the superior performance than traditional approaches for invariant multimedia information classification and retrieval, typically when the raw data is subject to linear transformations from different dimensions or with different dimensions in a multi-camera system.

Index Terms— bilinear invariants, information retrieval, dimensionality reduction

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 [1]. 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 Fig. 1 where the length of trajectory A is much longer than the length of 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 [2]. 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 Fig. 2, we have 18 trajectories in 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.
In functional analysis, the kernel of the general operator which could be non-linear is the set of all the operands \( v \) for which \( L(v) = 0 \). That is, if \( L : V \rightarrow W \), then

\[
\ker(L) = \{ v \in V : L(v) = 0 \},
\]

where 0 denotes the null vector in W, when the operator \( L \) is linear operator, the kernel space is also referred to as null space. For the transformation \( T \), the kernel space satisfies:

\[
T(L(v)) = 0,
\]

when \( L(v) = 0 \). In the following, we plan to show one typical example of computing the basis for kernel space invariants: bilinear invariants. Let us define:

\[
L(v) = AvB = 0,
\]

where \( A \) and \( B \) are \( m \times n \) and \( p \times q \) raw data matrices, the \( n \times p \) matrix \( v \) is the kernel space invariants.

**Lemma 1:** The bilinear invariant basis is invariant to the linear transformation on \( A \) and \( B \) simultaneously, when we apply the left-multiply linear transformation \( T_A \) on \( A \) and the right-multiply linear transformation \( T_B \) on \( B \). Typically, for \( AvB = 0 \),

\[
(B^T \otimes A)vec(v) = 0,
\]

where \( vec(v) \) is the vectorized matrix \( v \). If we denote \( M = (B^T \otimes A) \), where \( \otimes \) is denoted as Kronecker product, the constructive solutions for the \( i \)th basis for Bilinear Invariants (BI) is given by:

\[
v_1^{(i)} = det \begin{pmatrix} m_{1,1} & m_{1,2} & \ldots & m_{1,s_1} & m_{1,i} \\ m_{2,1} & m_{2,2} & \ldots & m_{2,s_2} & m_{2,i} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ m_{k,1} & m_{k,2} & \ldots & m_{k,s_k} & m_{k,i} \end{pmatrix}, \quad (5)
\]

\[
v_2^{(i)} = -det \begin{pmatrix} m_{1,1} & m_{1,2} & \ldots & m_{1,s_1} & m_{1,i} \\ m_{2,1} & m_{2,2} & \ldots & m_{2,s_2} & m_{2,i} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ m_{s,1} & m_{s,2} & \ldots & m_{s,s} & m_{s,i} \end{pmatrix}, \quad (6)
\]

\[
v_s^{(i)} = (-1)^{s-1}det \begin{pmatrix} m_{1,1} & m_{1,2} & \ldots & m_{1,s-1} & m_{1,i} \\ m_{2,1} & m_{2,2} & \ldots & m_{2,s-1} & m_{2,i} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ m_{s,1} & m_{s,2} & \ldots & m_{s,s-1} & m_{s,i} \end{pmatrix}, \quad (7)
\]

\[
v_k^{(i)} = 0, \forall k = s + 1, \ldots, t. \quad (9)
\]

When we apply the linear transformation on \( A \) and \( B \) simultaneously, we shall have

\[
(T_A A)v(BT_B) = (BT_B^T \otimes (T_A A)vec(v) = (T_B^T \otimes T_A)(BT_A^T \otimes A)vec(v) = 0, \quad (10)
\]

where the third equality is derived from the mixed-product property for Kronecker product. The above equation indicates that if we apply any linear transformation on the left of \( A \) and on the left of \( B \), the resulting space is invariant.

### 3. Tensor Representation for Bilinear Invariants

In the previous section, we have proposed the theoretical formulations of the bilinear invariants. Similarly, if there are more than two raw matrices which are subject to different linear transformations from different dimensions, we can generalize the tensor representations for bilinear invariants. Let us assume we have N raw data matrices \( A_1, A_2, \ldots, A_N \) where \( A_i \) has the dimension \( a_i \times b_i \), the tensor generalization for bilinear invariants can be represented as:

\[
X \times_1 A_1 \times_2 A_2 \times \ldots \times_n A_n = 0, \quad (11)
\]

where \( \times_i \) is denoted as \( i \)th mode product as described in [3]. Similarly, we can solve the tensor representation for bilinear invariants as follows:

\[
A_n^T \otimes (A_{n-1}^T \otimes \ldots \otimes A_1)vec(X) = 0 \quad (12)
\]

Once we obtained the equation (12), we rely on Lemma 1 to solve the invariant basis with similar approaches.
4. DIMENSIONALITY REDUCTION FOR BILINEAR INVARIANTS

As it can be seen from Section 2 and 3 that although matrix and tensor representation of bilinear invariants accurately provide the complete basis for invariants when linear transformations has been applied to different lengths or dimensions of features, the computational complexity of bilinear invariants is high. Here we propose the novel solutions of reduced dimension for bilinear invariants, we show that in the experiment, the reduced dimension for bilinear invariants can still achieve significantly high accuracy in multimedia information retrieval.

\[
AvB = 0, \\
\text{rank}(v) = w,
\]

where \(A, v, B\) are of the size \(m \times n, n \times p\) and \(p \times q\) respectively and \(w < \min(n, p)\).

**Lemma 2:** The complete solution for (13) is equivalent to the solution of the equation \(AXY B = 0\), where \(A, X, Y, B\) are of the size \(m \times n, n \times w, w \times p\) and \(p \times q\) respectively and \(X\) and \(Y\) are pair of invariant basis.

Now we solve the rank constraint problem as follows:

1. First, we solve the traditional null space invariants of \(A\) relying on \(AX = 0\) to obtain \(m - n\) basis for \(X = [X_1, X_2, \ldots, X_{m-n}]\).
2. Subsequently, we solve the null space invariants for \(B\) relying on \(Y B=0\) by first converting it into \(B^T Y^T = 0\) to obtain \(p - q\) basis for \(Y = [Y_1, Y_2, \ldots, Y_{p-q}]\).
3. We select \(w\) basis \(v = [v_1, v_2, \ldots, v_w]\) from \(X\) and \(Y\) to form the rank constraint invariant space by computing \(v_i = X_iY_i^T\).

5. SIMULATION RESULTS

5.1. Classification and Retrieval for Motion Events with Different Lengths of Features

We rely on Context Aware Vision using Image-based Active Recognition (CAVIAR) database [4] two dimensional motion trajectories to evaluate our classification and retrieval performance with Bilinear Invariants. For CAVIAR dataset, we select 3200 motion trajectories with different lengths including different human actions such as walking alone, meeting with others, window shopping, which are represented by the \(x\) and \(y\) coordinates. We further apply different linear transformations to different groups of trajectories by rotating each group of trajectories with different degrees due to different camera motions. Once we obtained the bilinear invariant representation for feature points 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) [5]. In Fig. 3, three precision-recall curves are plotted to demonstrate the efficiency and robustness of our approach. The curve relying on bilinear invariants has the best performance among the three curves compared to the approach with null space invariant and curvature scale space [2] with CAVIAR dataset. We also evaluate the performance of reduced dimension based bilinear invariants in retrieval in Fig.4. It can be seen from Fig.4 that even when we reduce the dimension of bilinear invariants to 40% of the original dimension, the retrieval performance is still comparable to null space invariants. We demonstrate the computational time for reduced dimension of bilinear invariants for retrieval in Fig.5.

5.2. Classification and Retrieval for Multimedia Database with Different Dimensions of Features

We also evaluate the performance of our approach for the classification and retrieval of multimedia information with different dimensions of features. We select 2547 three dimensional trajectories and 1296 two dimensional trajectories for the experiments from Australian Sign Language (ASL) dataset, where the 3D trajectories are obtained by extending the third dimension with the uniform time interval 0.1s. It has be to note that the null space invariants (NSI), curvature scale space (CSS) and centroid distance function (CDF) cannot deal with the retrieval and classification of multimedia information with different dimensions of features. Fig. 6 has demonstrated the classification accuracy for the mixture of 2d and 3d motion trajectories versus different number of class.

6. CONCLUSION

In this paper, we proposed a novel framework of Bilinear Invariants (BI) for affine invariant representation of ob-
The precision and recall curve for retrieval of multiple motion trajectories with different affine transformations and different lengths using reduced dimension bilinear invariants.

The computation time for retrieval with reduced dimensions with bilinear invariants.

The classification accuracy for motion trajectories with different dimensions versus different number of class.

<table>
<thead>
<tr>
<th>Number of Class</th>
<th>CSS</th>
<th>Bilinear Invariants</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 classes</td>
<td>0.731</td>
<td>0.866</td>
</tr>
<tr>
<td>25 classes</td>
<td>0.695</td>
<td>0.825</td>
</tr>
<tr>
<td>30 classes</td>
<td>0.676</td>
<td>0.801</td>
</tr>
<tr>
<td>35 classes</td>
<td>0.603</td>
<td>0.783</td>
</tr>
<tr>
<td>40 classes</td>
<td>0.584</td>
<td>0.769</td>
</tr>
</tbody>
</table>

7. REFERENCES


