Mining Semantics from Low-level Features in Multimedia Computing

by

Zheshen Wang

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree of
Doctor of Philosophy

Approved April 2011 by the
Graduate Supervisory Committee:

Baoxin Li, Chair
Gang Qian
Hari Sundaram
Jieping Ye

ARIZONA STATE UNIVERSITY

May 2011

ABSTRACT

Bridging semantic gap is one of the fundamental problems in multimedia computing and pattern recognition. The challenge of associating low-level signal with their high-level semantic interpretation is mainly due to the fact that semantics are often conveyed implicitly in a context, relying on interactions among multiple levels of concepts or low-level data entities. Also, additional domain knowledge may often be indispensable for uncovering the underlying semantics, but in most cases such domain knowledge is not readily available from the acquired media streams. Thus, making use of various types of contextual information and leveraging corresponding domain knowledge are vital for effectively associating high-level semantics with low-level signals with higher accuracies in multimedia computing problems.

In this work, novel computational methods are explored and developed for incorporating contextual information/domain knowledge in different forms for multimedia computing and pattern recognition problems. Specifically, a novel Bayesian approach with statistical-sampling-based inference is proposed for incorporating a special type of domain knowledge, spatial prior for the underlying shapes; cross-modality correlations via Kernel Canonical Correlation Analysis is explored and the learnt space is then used for associating multimedia contents in different forms; model contextual information as a graph is leveraged for regulating interactions among high-level semantic concepts (e.g., category labels), low-level input signal (e.g., spatial/temporal structure).

Four real-world applications, including visual-to-tactile face conversion, photo tag recommendation, wild web video classification and unconstrained consumer video summarization, are selected to demonstrate the effectiveness of the approaches. These applications range from classic research challenges to emerging tasks in multimedia computing. Results from experiments on large-scale real-world data with comparisons to other state-of-the-art methods and subjective evaluations with end users confirmed that the developed approaches exhibit salient advantages, suggesting that they are promising for leveraging contextual information/domain knowledge for a wide range of multimedia computing and pattern recognition problems.

ACKNOWLEDGEMENT

First and foremost, I would like to thank my advisor, Professor Baoxin Li for being a great mentor. As a research scientist, his great passion on researches and high spirit of enterprise and hardworking set an admirable exemplar to all his students. As a professor and graduate advisor, he never hesitates spending time and sharing his experience and opinions in guiding students to shape their own views. I appreciate his inspiring encouragement to my Ph.D life, his elaborate guidance and constructive feedback on my work, and his patience in helping me improve my professional skills from different aspects. The conversations with him about my future career and life are also very beneficial. Thanks, Dr. Li, for all your support and help in the past five years!

I would like to thank my committee members, Professor Gang Qian, Professor Hari Sundaram, and Professor Jieping Ye for serving as members of my dissertation committee and providing enlightening discussions and suggestions.

I would like to thank my colleagues at Arizona State University, as well as collaborators at Google Research and Kodak Research Laboratories. In particular, I would like to thank my previous and current group fellows, Dr. Xinyu Xu, Dr. Jin Zhou, Dr. Wenfeng Li, Xiaolong Zhang, Qiang Zhang, Nan Li (ASU), Naveen Kulkarni, Devi Archana Paladugu, the close interactions between us really make our group like a big loving family. I would also like to express my sincere thanks to Liang Sun and Jianhui Chen for their constructive discussions on my researches. I appreciate the great help from Dr. Ming Zhao, Dr. Sanjiv Kumar, Dr. Mrityunjay Kumar and Dr. Jiebo Luo, who made my Fall 2009 and Summer 2010 internship valuable and memorable experiences.

My heartfelt appreciations go to my dear parents Wenbin Wang and Chunfeng Lin and uncle Jiguo Lin. Their love and unconditional support throughout all the years of my life so far made my Ph.D studies possible. I am also deeply grateful to the great couple Elizabeth Mann and Fredric Mann, and my dear friends Nan Li (ASU, CMU), Liu Wei

and other fellow students and friends at ASU. Friendships with them made me feel ASU and Phoenix is my second hometown.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

Chapter 1

INTRODUCTION

1.1. Background and Motivation

With a rapid increase of accumulating digital media collections, content processing and semantic analysis have emerged as an important yet challenging problem in multimedia computing and pattern recognition. However, there exists a huge data-meaning gulf from low-level computational representation (e.g., pixels) towards high-level semantic interpretation of the perceived contents which users are expecting. This is well known as *semantic gap*. Unfortunately, semantics are often conveyed implicitly in the context (i.e., information surrounding the object/event that helps to determine its interpretation), relying on interactions among multiple levels of concepts or low-level data entities. Without a constraint of context, associations between low-level features and high-level semantic concepts are often unrecoverable or ambiguous. For example, in a face image, there is no way to tell from any single pixel that this image depicts a human face; In video understanding, if the temporal order of frames is randomly scrambled, it is difficult to distinguish "stand up" and "sit down" actions. Also, additional domain knowledge may often be indispensable for uncovering the underlying semantics, but in most cases such domain knowledge is not readily available from the acquired media streams. For example, presence of red roses and white candles in an image is a strong indication of a wedding. However, even roses and candles can be successfully detected (which is non-trivial in reality), associating the concept "wedding" to the image is not an easy task without knowing the semantic connections between such an abstract word and visual appearances of the physical objects.

In the past few decades, much effort has been made for closing semantic gap in multimedia computing. Existing techniques can be mainly divided into three categories:

generating computational representations (i.e., features) of perceived contents, measuring similarities/distances between visual objects and establishing associations between the obtained data representation and the targeted description (i.e., semantics). All of these three tasks can be performed on hierarchical layers towards different semantic levels (e.g., region, object, scene, event, etc.). We briefly review the related work in the below. While the following review is by no meaning to be exhaustive due to space limitation, we attempt to cover those research activities or existing systems that are closely related to the proposed approaches.

Feature detection is a fundamental issue for any pattern recognition problems. A large variety of features have been investigated for interpreting low-level multimedia data [133-134]. Popular features include local/regional features, such as color, texture, which usually forms a bag-of-word representation of raw data for further processing; global or accumulating features computed from the entire object, such as histogram or concatenated local/regional features, which reflects global statistics or spatial/temporal layout conveyed in the original data. In addition, middle-level shape and object features, such as straight lines, human faces, are often extracted for specific applications.

With an appropriate computational representation of raw data, similarity measures are then used for further comparison, grouping and discrimination. Widely employed metrics include basic distance measures (e.g., Euclidean distance, cosine distance, Mahalanobis distance etc.) which are typically used for vectorized features; more complicated metrics for shape/object features (such as Scale Hausdorff distance [1], context based shape matching [2]); and advanced structural features in which numerical descriptors are hierarchically ordered and the order is taken into consideration in computing the similarities/distances, such as edit distance [3], graph-based matching method [4], etc.

For finally linking a feature representation to the corresponding semantics, rule-based methods and learning based approaches are applied. Statistical learning algorithms have become the main trend to this problem in recent years. Prevalent methods include Bayesian approaches (e.g., Naïve Bayesian approach [5]), probabilistic graphical models (e.g., Conditional Random Fields [76]), latent space analysis methods (e.g., Canonical Correlation Analysis [57], Sparse Coding techniques [94]), kernel-based learning and inference framework, etc.

The above techniques provide us with basic tools for mining semantics from low-level observables. However, in order to make real-world problems tractable, models are often under assumptions of independencies among data or semantic concepts. As mentioned previously, multimedia data exhibit strong spatial and temporal structures and the corresponding semantic concepts are rarely independent as well. Thus, leveraging the widely existing contextual information and related domain knowledge is crucial to achieve good performance in interpreting low-level signals from high level.

Efforts have been devoted to modeling and utilizing contextual information/domain knowledge in different forms [75, 76, 126-132]. Media streams (e.g., image, videos) often exhibit long-range dependencies. However, direct modeling of global constraints/interactions becomes computationally intractable even for a small piece of media object (e.g., a small image/shot video). This paradox can be resolved to a large extent by graphical models [75, 76, 128, 132], as it is relatively easier to encode the structure of local dependencies from which we would be able to achieve global consistencies. When modeling contextual interactions/domain knowledge for media objects, it is important to take into consideration data variations and other uncertainties due to noises. This naturally leads to a probabilistic framework of computational algorithms, in which the final predictions of high-level semantics can be seen as inference with respect to some cost function. Graphic models are often combined with a probabilistic framework, which is known as Probabilistic Graphic Models. However, not all contextual information/domain knowledge can be inferred in a parametric way;

sometimes, it cannot be explicitly conveyed as a graph. In these cases, non-parametric methods and numerical analysis are often utilized. For example, in multi-tasking learning [126-131], connections among different tasks can be leveraged via imposing a regularization term or a low-rank constraint on a joint objective function [129, 131] or combined multiple pre-specified kernel matrices [130].

1.2. Proposed Approaches

In this work, we propose to leverage contextual information/domain knowledge in a single modality or between different modalities for effectively associating high-level semantics with low-level signals with higher accuracies in multimedia computing problems. Given contextual information/domain knowledge *R*, we seek a mapping *f* which associates low-level signals and high-level concepts of media objects:

$$C = f(X), \text{ subject to } R \tag{1.1}$$

where $X \in \mathbf{X}$ refers to low-level features of media objects **O**, e.g., image, video, text and **X** is the feature space; $C \in \mathbf{C}$ denotes high-level semantic concepts and their configurations e.g., a set of category labels and their semantic relationships, and **C** is the space of semantics; $R \in \mathbf{R}$ denotes a variety of contextual information/domain knowledge in *X*, *C* or in-between, such as cross-modality correlations, hierarchical taxonomy, spatial structure, temporal order and other domain-specific priors.

In this work, we consider three ways of incorporating contextual information/domain knowledge *R* in uncovering the mapping *f*, including incorporating *R* as statistical prior under a Bayesian framework, capturing and formulating *R* by exploring correlations between different modalities, and modeling *R* as a graph for regulating interactions among *X* or *C*. These three methods are briefly introduced in the following subsections.

1.2.1 Incorporate contextual infromation as statistical prior under a Bayesian framework

If contextual information/domain knowledge $R \in \mathbf{R}$ can be formulated as a probability of the occurrence of high-level concepts $C \in \mathbf{C}$, such as

$$R = p(C) \tag{1.2}$$

it can be naturally incorporated as a statistical prior under a Bayesian framework in inferring the association between low-level observations X and high-level semantic concepts C:

$$p(C \mid X) = \frac{p(C)p(X \mid C)}{p(X)} \propto p(C)p(X \mid C) \tag{1.3}$$

$$\hat{C} = \arg\max_{C \in \mathbf{C}} p(C \mid X) \tag{1.4}$$

```
(1) Generate $L$ random samples based on $p(C)$, $C_1,\ldots, C_L$
(2) Loop until a certain stop criterion fulfilled:
    (2.1) Compute likelihood for each sample $p(X|C_i)$, $i = 1,\ldots L$;
    (2.2) Re-sample proportional to the likelihood.
(3) Compute the final model from the weighted samples.
```

Figure 1: Proposed statistical sampling algorithm.

For real-world applications, the density is typically multi-modal and the dependency model on observations X is highly nonlinear. Thus using a parametric form for the density would be challenging. Consequently, we propose to use a statistical-sampling-based algorithm for the estimation problem, as done in [33]. At the beginning, the samples are drawn around the parameters initialized from prior $p(C)$. The samples will then be updated iteratively based on the given observations, which leads to a particle-filtering-like scheme as summarized in Figure 1. Random sampling process involved in this approach alleviates the risk of local optimum and the prior information provides effective constraints for regulating the ranges of sampling, which contributes to a more efficient learning process and more accurate results.

1.2.2 Model and capture correlations between different modalities via KCCA

Contextual information is often implicitly conveyed in the input data, for example, the underlying correlations between two sets of observations from different modalities of the same media objects. Canonical Correlation Analysis (CCA) [57] attempts to find basis vectors for the two sets of variables such that the correlation between the projections of the variables onto these basis vectors is mutually maximized. The correlation between the two sets of variables may not be visible in their original coordinate system. CCA finds a linear transformation for two sets variables such that in the transformed space they are maximally correlated. The canonical correlation between any two sets of variables (e.g., from different modalities) is defined as

$$\rho = \max_{R_x, R_y} corr(F_x \cdot R_x, F_y \cdot R_y) \tag{1.5}$$

where F_x and F_y are the two sets of variables, and R_x and R_y are the basis vectors onto which F_x and F_y are projected, respectively. The problem of finding ρ is therefore an optimization problem with respect to R_x and R_y. This optimization problem can be formulated as a standard eigen analysis problem [56] which can be solved with standard methods. Since R_x and R_y are always calculated to maximize the correlation of the projections, CCA is independent of the original coordinate system unlike other correlation analysis techniques. There may be more than one canonical correlation, each representing orthogonally separate pattern of relationship between the two sets of variables. The canonical weights represent the unique positive or negative contribution of each variable to the total correlation.

When two sets of multi-dimensional variables from different modalities are available for the same media objects, for example, low-level features $X = [X_1, X_2, ..., X_n]$ and high-level semantic concepts $C=[C_1,C_2,...,C_n]$ can be viewed as two sets of variables from different modalities for n training media objects $O=[O_1,O_2,...,O_n]$. The projections found by CCA can be thought of as capturing the underlying correlations between the two modalities

$$R=[\hat{R}_X,\hat{R}_C]=\arg\max_{R_X,R_C} corr(X\cdot R_X, C\cdot R_C) \tag{1.6}$$

The obtained R can be further used to infer the associations between low-level features and their high-level semantics concepts of a new media object O_0 by selecting the concepts C_i from the training set whose corresponding features X_i is closest to the features X_0 of the input object O_0:

$$\hat{C}=C_i,\ i=\arg\min_{i=1,...,n} dist(X_0\cdot\hat{R}_X, X_i\cdot\hat{R}_X) \tag{1.7}$$

To captured non-linear correlations, input variables can be first mapped onto a pre-defined kernel space, in which CCA is then performed.

1.2.3 Model contextual infromation as graph for regulating interactions among low-level features, high-level semantic concepts or in-between.

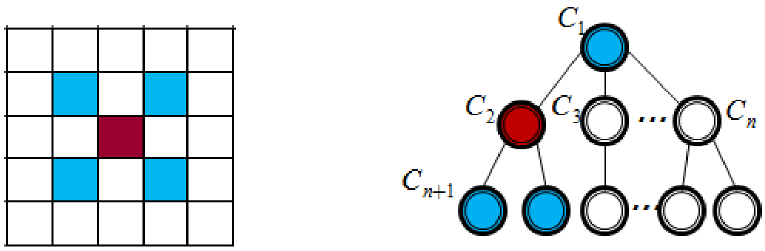
Other than utilizing the underlying correlations between different modalities, it is also possible to model contextual information/domain knowledge among low-level features, high-level concepts as graphs. Such information is powerful for improving accuracies in interpreting multimedia data if it is utilized in a proper way. In this work we explore a parametric graph model -- Tree- Discriminative Random Field (Tree-DRF) and a non-parametric method -- Sparse Representation based on Weighted-Sequence Distance kernel (WSD kernel) for incorporating graphs of domain knowledge in inferring the mapping from low-level signals to high-level semantic concepts.

1.2.3.1 Tree- Discriminative Random Field (Tree-DRF)

Conditional Random Fields (CRFs) are graph-based models that are popularly used for labeling structured data such as text [76] and were introduced in computer vision by [75]. We denote the observations as X and the corresponding concepts as C. Given knowledge as a graph $R = \{S, N\}$, where S is the set of nodes and N_i is the set of neighbors of node i, the conditional distribution over concepts given the observations is defined as a Gibbs field:

$$p(C \mid X) = \frac{1}{Z} \exp(\sum_{i \in S} A_i(C_i, X) + \sum_{i \in S} \sum_{j \in N_i} I_{ij}(C_i, C_j, X)) \tag{1.8}$$

where Z is a normalizing constant called partition function. Terms A_i and I_{ij} are the unary and pariwise potentials sometimes referred to as *association potential* and *interaction potential* respectively.



(a) (b)

Figure 2: Possible graphs used in DRF (Red node—current node; blue nodes—neighbors of the current node): (a) Grid structure considered in [75]; (b) Tree structure used in this work.

In [75], the graph is built on images, which forms a symmetric grid (as illustrated in Figure 2-(a)), so that the weighs for neighbors are homogeneous. In this work, we impose the graph on the taxonomy of high-level semantic concepts, which follows a hierarchical tree structure. Numbers of neighbors for each node may vary, thus the weights for different neighbors become non-homogeneous.

Standard maximum likelihood method is used for parameter learning in Tree-DRF. Since the graphical structure is a tree, exact unary and pariwise marginals were computed using Belief Propagation (BP). Given a training set which consists of *n* samples, optimal parameters can be computed as

$$[\hat{A},\hat{I}] = \arg\max_{A,I} \sum_{k=1}^{n} p(C^{(k)} \mid X^{(k)}) \tag{1.9}$$

For inference, we used sitewise Maximum Posterior Marginal (MPM), again using BP. Concepts of test sample X_0 can be inferred with the learnt parameters as

$$\hat{C} = \arg\max_{C\in\mathbf{C}} p(C \mid X_0) = \arg\max_{C\in\mathbf{C}} \frac{1}{Z}\exp(\sum_{i\in S}\hat{A}_i(C_i, X_0) + \sum_{i\in S}\sum_{j\in N_i}\hat{I}_{ij}(C_i, C_j, X_0)) \tag{1.10}$$

1.2.3.2 Sparse Representation Based on Weighted-Sequence Distance Kernel

In addition to the parametric approach, it is also possible to incorporate contextual information in a non-parametric way. In this work, we propose to represent, compare and reconstruct video sequences under a sparse representation framework in which global temporal order of the sequence is incorporated. Obtained sparse coefficients can be further used to extract high-level semantics from the input videos, e.g., extract a temporally-compressed summary video with respect to both of the visual contents and sequential order of the original sequence. The proposed approach is briefly introduced in the below.

Sparse representation aims at computing linear sparse coefficients with respect to an over-complete dictionary of a set of basis elements [94]. Suppose we have an underdetermined system of linear equations:

$$y = A \cdot \alpha \tag{1.11}$$

where $y \in R^m$ is the target signal to be approximated, $\alpha \in R^n$ is the vector for unknown reconstruction coefficients, and $A \in R^{m\times n}$ (m<n) is the over-complete dictionary with *n* bases. Generally, a sparse solution is more robust and efficient for coding and

reconstructing the target signal and has been widely used for various vision related applications, such as image restoration [95], The sparsest solution can be obtained by solving a L_1 optimization problem in polynomial time by standard linear programming method [96]:

$$\min_{\alpha} \| \alpha \|_1, \; s.t. \; y = A \cdot \alpha \tag{1.12}$$

In this work, the input video sequence y that is to be analyzed (i.e., the target signal) is represented as

$$y = [f_1, f_2, ..., f_n]^T \tag{1.13}$$

in which f_i ($1 \leq i \leq n$) is a short snippet (defined as a group, e.g., 5-10, of consecutive frames from the original video). For the sake of simplicity, here we call it a *snippet*. The contextual information R is the original sequential order of the frames, which is retained in the short snippets and among the relative order of all snippets as well.

For reconstructing a video by a small set of frames/snippets from the input video with the original order, we first generate a dictionary A of M elements

$$A = [a_1, a_2, ..., a_M] \tag{1.14}$$

in which each dictionary element a_j is a subset of snippets selected from y

$$a_j = [x_1, x_2, ..., x_n]^T, \; 1 \leq j \leq M \tag{1.15}$$

and a_i contains exactly l non-zero entries (i.e., l snippets from the original video),

$$x_i = \begin{cases} f_i, \; i \in S_j \\ 0, \text{ otherwise} \end{cases}, \; 1 \leq i \leq n, \; |S_j| = l \tag{1.16}$$

where S_j denotes the set of l indices derived from the input video y to construct j^{th} dictionary element a_j. We represent y_k as a sparse linear combination of dictionary elements as shown below:

$$y_k = A \cdot \alpha, \; \|\alpha\|_0 = m, \; m << M \tag{1.17}$$

where α is an m-sparse coefficient vector, i.e., only m non-zeros entries are allowed in

α. y_k is a linear combination of all dictionary elements in A with non-zeros coefficients while retaining their temporal order (i.e., contextral constraint).

The sparse coefficient vector, α, is estimated by minimizing the error between y and y_k as given below:

$$\hat{\alpha} = \arg\min ErrFn(y, y_k), \quad \|\alpha\|_0 < m \tag{1.18}$$

where $ErrFn(\cdot,\cdot)$ compares the two inputs and estimates the error. Typically, L_2-norm is used in such cases. But, in this case, $ErrFn(\cdot,\cdot)$ needs to be selected carefully as y and y_k are sequences of different lengths rather than regular vectorized data points, standard L_2 norm is no longer applicable for computing the reconstruction error here.

To solve this problem, we propose Weighted-Sequence Distance kernel (WSD kernel), a generalized version of classical Levenshtein Distance (also known as String Edit Distance [120]), which computes the distance of two sequences of different lengths as the total cost for converting one sequence into the other with editing operations, such as *Insert*, *Delete*, *Copy/Substitute*. Global order of the sequences is imposed in the process of searching for the optimal operation procedure by using dynamic programming. Due to the fact that frames in video sequences are not distinct, classical edit distance which was designed for characters does not applicable since each character is treated equally in the original formulation. To compensate for this drawback, we first cluster raw video frames and create a codebook of representative frames (e.g., cluster centroids); the original sequences are then coded as a sequence of codeword-weight pairs (denoted as super-frames) in which the weight is the normalized number of consecutive frames which are assigned to the same codeword. The coded weighted-sequences are further used for computing edit distance from one sequence to the other. In the proposed formulation, not only the codeword, but also the weights and the pariwise similarities between codewords contribute to the total cost of edit operations. We shall discuss more details in Section 5.3.3.

The obtained sparse coefficients $\hat{\alpha}$, which best reconstruct the input video based on dictionary A given sparsity m, capture the underlying semantics of the video including salient snippets $\hat{a}_j$ and sequential order of frames in or between snippets.

$$\hat{C} = \{\hat{a}_j \mid \hat{\alpha}_j \neq 0,\ 1 \leq j \leq M\} \tag{1.19}$$

where $\hat{a}_j$, which are corresponding to the non-zeros $\hat{\alpha}_j$ coefficients, are kept (respecting the original order of the indexes) as high-level semantics for describing the major contents of the entire video.

1.2.4 Selecting Appropriate Methods for Different Scenarios

The above methods can be applied when different types of contextual information/domain knowledge are available. Specifically, when contextual information/domain knowledge can be expressed in a probabilistic form, it can be naturally incorporated as prior under a Bayesian framework; when multiple types of observations are available for the same source of multimedia objects, the underlying correlations can be uncovered by using KCCA and further used as contextual information for predicting high-level semantics for new data entities; in addition, if contextual information/domain knowledge can be represented as a graph, parametric or non parametric methods based on graph mode can be considered for leveraging such information.

1.3. Contributions

In this dissertation, we explore three different ways of incorporating contextual information/domain knowledge for multimedia computing and pattern recognition problems. Four applications were selected for demonstrating the effectiveness of the proposed approaches. Major contributions of this work are summarized in the below.

We propose a novel Bayesian Active Shape Model (BASM) with a statistical sampling based parameter learning scheme for 2D face shape alignment, in which domain knowledge (i.e., anthropometric face statistics) is incorporated over an anchor point based 2D face model. The imposed prior information effectively regulates sampling ranges of the model parameters, which contributes to improved accuracies of face alignment over existing approaches.

We propose to uncover the underlying semantic correlations between visual contents and text tags of Flickr photos and use the obtained information for recommending tags for new images. Compared to state-of-the-art work from Yahoo, this approach does not rely on any tag from the user and the resulted accuracies on real Flickr photos are superior.

We propose novel Tree-DRF fusion approach for categorizing wild web videos in which a predefined taxonomy tree (i.e., a graph) is incorporated in the process of predicting multi-class category labels for web videos. This approach is effective for combining training data from multiple sources and is robust for filtering out noises. It supports multi-class classification for single videos and is able to achieve global optimal category label assignments over all categories. It significantly outperforms commonly used fusion strategies based on SVM and iterative co-training approaches on a large-scale data set of 80K Youtube videos of unconstrained contents.

We propose a novel approach for analyzing video contents via sparse representation based on novel Weighted-Sequence Distance kernel (WSD kernel). Different from regular sparse representation methods, we explicitly retain the temporal order of the input sequence in forming the dictionary, computing reconstruction errors and obtaining sparse coefficients for reconstructing the original video. Application to content-based consumer video summarization confirms the effectiveness of leveraging sequential structure in such a non-parametric way.

Experiments were performed on four applications ranging from classic research challenges to emerging tasks in multimedia computing. These applications are highly correlated in the challenge of bridging a persistent gap between low-level signals and high-level semantics through leveraging the contextual information/domain knowledge, which shows the broad impact of the topic and the generality of the proposed solutions.

1.4. Thesis Organizations

In this dissertation, we explore three ways and develop specific algorithms to incorporate contextual information/domain knowledge in solving real-world multimedia computing problems. The remainder of this dissertation is organized as follows.

In Chapter 2, a novel Bayesian Active Shape Model (BASM) is proposed for leveraging anthropometric face statistics in 2D face shape alignment. Both domain priors and spatial constraints are considered for extracting high-level semantic and repurposing the information in an alternative form (i.e., tactile representation).

In Chapter 3, we capture cross-modality correlations between visual images and their corresponding textual tags via KCCA and make use of the underlying connections for automatically recommending tags for Flickr photos.

We further explore parametric and non-parametric approaches for modeling contextual information/domain knowledge as graphs. Chapter 4 describes a novel Tree-DRF approach, in which a pre-defined taxonomy structure of high-level concepts is used for regulating multi-label multi-class classification of YouTube videos. Chapter 5 presents a novel sparse representation approach based on Weighted-sequence Distance Kernel (WSD kernel) for consumer video summarization, where temporal order of video frames is naturally incorporated in sequence comparisons and sparse representation using subsequences.

Chapter 6 summarizes the dissertation with conclusions drawn from our current work and potential possibilities of future researches along the direction.

Chapter 2

USING SPATIAL PRIOR UNDER A BAYESIAN FRAMEWORK FOR VISUAL-TO-TACTILE FACE CONVERSION

Portrait photos (facial images) play important social and emotional roles in our life. This type of visual media is unfortunately inaccessible by users with visual impairment. This section proposes a systematic approach for automatically converting human facial images into a tactile form that can be printed on a tactile printer and explored by a user who is blind. We propose a deformable Bayesian Active Shape Model (BASM), which integrates anthropometric priors with shape and appearance information learnt from a face dataset. We design an inference algorithm under this model for processing new face images to create an input-adaptive face sketch. Further, the model is enhanced by input-specific details through semantic-aware processing. We report experiments on evaluating the accuracy of face alignment using the proposed method, with comparison with other state-of-the-art results. Furthermore, subjective evaluations of the produced tactile face images were performed by seventeen persons including six visually-impaired users, confirming the effectiveness of the proposed approach in conveying via haptics vital visual information in a face image.

2.1. Background and Overview of the Proposed Approach

Digital visual information in graphical forms (e.g., digital images, maps, diagrams, etc.) has become prevalent in the information era and the sighted people can easily enjoy the added value of graphical contents. Unfortunately, people with visual impairment are partially or totally deprived of this benefit. Although modern computer technologies have provided various text-to-Braille/audio solutions that enable convenient access to text, computer users with visual impairment still cannot access graphical contents without the assistance of sighted people. The typical procedures for manually producing tactile graphics by sighted professionals are in general time-consuming and labor-intensive, and hence the coverage is extremely limited. Further, there is no on-demand and independent availability if the production has to be done by third-party professionals. What are still missing are automatic approaches that support real-time and independent access to graphical contents by users with visual impairment.

Studies along this direction typically need to first address the fundamental issue of image simplification, due to the extremely limited bandwidth of tactile perception compared with that of vision. Existing work relies on either computer-aided manual processing (e.g., using drawing software) or simple image processing steps such as edge detection. Despite the existence of some initial attempts [11-13], this fundamental step towards automating the creation of tactile graphics from images remains to be largely unsolved. One prominent challenge is that low-level image processing techniques such as edge detection cannot ensure to retain semantically meaningful information, especially if the techniques are expected to work for any types of graphics. For example, broken and scattered edge segments may serve only to confuse a blind user if they are directly mapped to tactile lines; and attempts to clean up the edges, such as linking short ones to form a long contour, may do harm if those processing steps are purely driven by the data.

In this work, we limit the scope of our study to a special type of graphic, human facial images, for the special value that they have in a person’s social and emotional life. We aim at developing a systematic approach to automatic conversion of a human face image into its tactile form. Limiting the scope to this special type of images enables us to introduce higher-level semantics for guiding lower-level image processing steps in designing robust algorithms for automated visual-to-tactile conversion. Exploiting the constraints imposed by knowing that the image contains a face, we first propose a deformable Bayesian Active Shape Model (BASM), which integrates anthropometric facial priors with both shape and appearance information learnt from a face dataset, for modeling human faces. Then a statistical-sampling-based inference procedure is introduced under the model, for obtaining a data-adaptive version of the model for any given face image. Serving as a starting point, this model enables additional semantic-aware processing steps that are designed to enrich the sketchy face model with more input-specific details, resulting in the final tactile face images. As such, the proposed approach combines anthropometric prior knowledge, learnt model generality and given data specificity to automatically create an informative tactile representation of the original face image. Such a tactile representation can be readily rendered by a tactile printer, and thus potentially provide a desired solution to the problem of creating on-demand tactile faces independently by a user with visual impairment. Figure 3 illustrates the overall processing flow of the proposed approach.

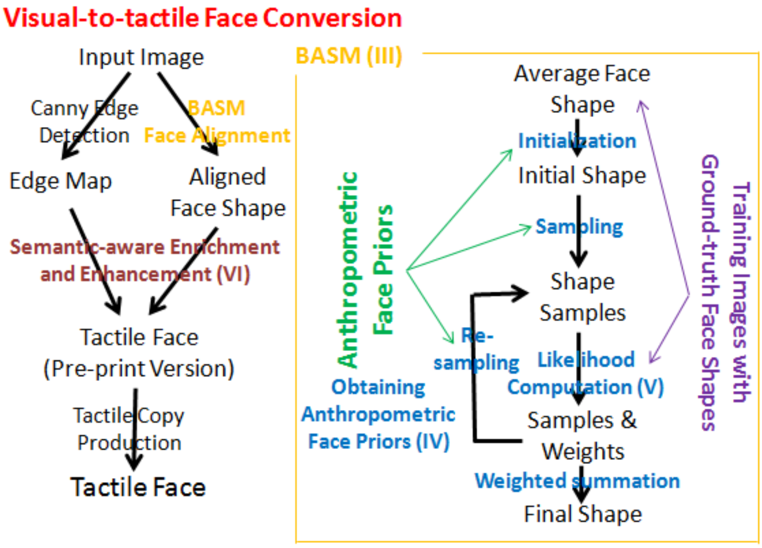

Visual-to-tactile Face Conversion
Input Image
Canny Edge Detection
BASM Face Alignment
Edge Map
Aligned Face Shape
Semantic-aware Enrichment and Enhancement (VI)
Tactile Face (Pre-print Version)
Tactile Copy Production
Tactile Face
BASM (III)
Anthropometric Face Priors
Obtaining Anthropometric Face Priors (IV)
Average Face Shape
Initialization
Initial Shape
Sampling
Shape Samples
Re-sampling
Likelihood Computation (V)
Samples & Weights
Weighted summation
Final Shape
Training Images with Ground-truth Face Shapes

Figure 3: Overall processing flow of the proposed approach (corresponding section number are specified in the parentheses).

In the following subsections, we review related work in Section 2.2. In Section 2.3, then present the proposed visual-to-tactile face conversion approach based on a novel Bayesian Active Shape Model, followed by details of obtaining anthropometric face priors in Section 2.4 and computing of the shape likelihood in Section 2.5. Semantic-aware enrichment steps for creating the final tactile face images are described in Section

2.6. We present face alignment results with comparisons to other state-of-the-art methods and report systematic user evaluation of the outputs produced by our approach in Section 2.7. Section 2.8 concludes the section with brief discussion on future work.

2.2. Related Work

Conventional approaches to manual creation of tactile graphics involve many tedious tasks [14] that are time-consuming and labor-intensive. Automated approaches to visual-to-tactile conversion have been the focus of some recent studies. Some existing work can only handle simple line-drawing graphics (e.g., [15, 16]), with little effort dealing with acquired images such as portrait images that we attempt to address in this work. For acquired images, the work of [11] relied on simple image processing steps such as negation and edge detection, and Way et al [12, 13] proposed to simplify images mostly by edge detection. The system developed in [17] resorts to Photoshop for image simplification, which still requires some manual efforts from a sighted person and thus does not address the need of an automated solution. In [18], a multi-modal approach was proposed to present digital graphics. Although the concept of semantics-aware processing was introduced to modulate the edge detection step, refined solutions to handle specific types of graphics remain to be developed.

Since we focus on human face images in this work, we also briefly review recent face alignment work in the below, which is a key component in our approach. Face alignment is an active research area with many research papers in recent years. In the pioneering work of Active Shape Model (ASM) [19], the contours of major facial features are represented by a set of feature points, and in matching a model to a given image, feature points are updated iteratively by searching along profiles around the current positions and fitting to a set of model parameters. Bayesian Tangent Shape Model (BTSM) [20] is another derivation of ASM proposed to infer shape parameters by the EM algorithm. While being useful, ASM may suffer from the local minima problem if the optimization of shape points is based on a gradient decent search scheme [21]. To alleviate this problem, a hierarchical CONDENSATION approach was discussed in [22] to search the MAP estimates of shape configurations.

Deformable model based approaches often encounter difficulties in achieving desired specificity to a particular instance while retaining enough model generality. A common remedy is to impose prior knowledge on possible shape deformation. Typical prior information utilized is linear shape deformation subspace learnt from training images (e.g., [29, 30]). Nevertheless, it often strongly restricts the deformation and biases towards the training set. A number of approaches have been proposed to remedy this problem. Kernel PCAs were proposed to extend the linear PCA subspace in order to retrieve more shape variations [23, 24]. Huang et al [21] created separated deformable models for each face component and use a probability distribution function to encode the interrelationship among parameters of all modeled components by constrained Gaussian Process Latent Variable Model. Liang et al [25] integrated the Markov Network search with the global shape prior to improve the alignment. Gu et al proposed a shape regularization model, which incorporates non-linear shape prior from a mixture of constrained Gaussian components with extra noises [26]. While improved robustness for exaggerating expressions and large occlusions was shown, shape priors in the work are still purely built upon the training set. Other forms of prior information include generic properties of local curves (e.g., continuity and smoothness) [27]. There also approaches that use fewer number of features points (10 to 20) [39, 41], which do not provide desired level of detail for tactile conversion. Other contributions to the face alignment problem include [28-32], which are less relevant to the focus of our task.

In our recent work [33] along the same direction of this study, we reported preliminary results with a simpler approach which did not take into consideration the anthropometric priors in the modeling. Consequently, the results were not as good as desired albeit encouraging. Also, the evaluation of [33] was preliminary, with only one blind user. In this work, we propose to use domain knowledge of the shape, i.e., anthropometric face constraints, as the prior in developing the Bayesian Active Shape Model. Such a prior reflects common biological features of human faces and thus is potentially useful for providing desired constraints in generating physically-meaningful shapes from a generic model. We also report experiments of more comprehensive evaluation with 17 people including 6 visually-impaired persons.

2.3. The Proposed Approach

In this section, we present the proposed approach for creating tactile facial images automatically. This approach consists of three major steps. We first model human faces using a novel Bayesian Active Shape Model (BASM), in which a deformable shape model of human faces is first learnt from a training set, with prior anthropometric constraints incorporated in the model. Then, given a test face image, the set of model parameters that best explains the image data is estimated through Bayesian inference with statistical sampling approach. With the face model and the input image aligned, we further employ a semantic-aware processing step to enrich the sketchy model in producing the final tactile face image. These three steps are described in the following subsections A to C respectively, with elaborated details presented in Sections 2.4, 2.5 and 2.6.

2.3.1 BASM for Face Modeling

Active Shape Model (ASM) is widely used for modeling landmark-based shapes. However, traditional ASM suffers from two major drawbacks. First of all, ASM shape model is purely built on training images. Furthermore, in ASM, all variations are jointly captured by eigen vectors and eigen values of the training data, which makes it difficult to manipulate parameters to generate desired shapes corresponding to specific facial expressions and/or poses.

Although human face shape varies among different people, the variations are bounded by biological constraints that can be estimated by anthropometric measures of the head and face. Such strong domain knowledge can be used as prior information for constraining shape generalization. Based on this, we propose a novel Bayesian Active Shape Model (BASM) that embeds prior knowledge of anthropometric face measurements into an active shape model. Further, separate parameters for scaling, rotation and local shape variations are explicitly defined in the proposed BASM so that the deformable model can be more accurately controlled with parameters that are physically intuitive.

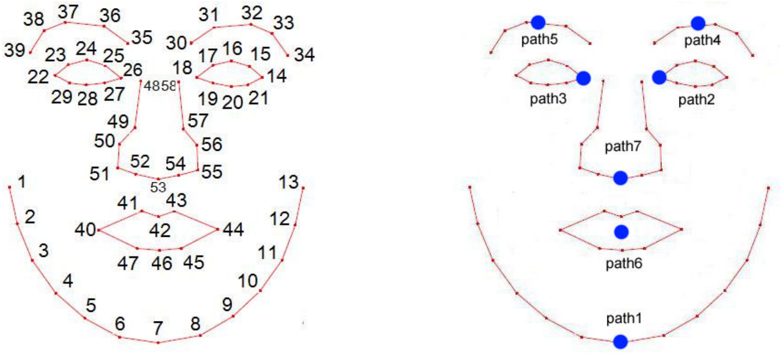


Figure 4: Left: Anchor point based face model; Right: Point-paths and corresponding key points.

We start with an anchor points based face shape model as in [19], where human faces are characterized by N anchor points, $\mathbf{p}_i=(x_i, y_i)$, $i=1,...N$. Specifically, we adopt the 58-anchor point model from [34], where major facial contours are captured by 58 landmarks around the eyebrows, the eyes, the nose, the mouth, and the chin/jaw (Figure 4-Left). Coordinates of all facial landmarks are denoted as $\mathbf{f}_i = [x_{i1},y_{i1},x_{i2},y_{i2},...,x_{iN},y_{iN}]$. The objective of shape modeling is to form a parameterized model for representing any face shape $\mathbf{f}$ from a basic shape $\mathbf{f}_0$ by varying a limited number of parameters, i.e.,

$$\mathbf{f} = \psi(\boldsymbol{\theta}, \mathbf{f}_0) \tag{2.1}$$

where $\boldsymbol{\theta}$ is the set of parameters of the shape model.

In traditional ASM [19, 34], Principle Component Analysis (PCA) is directly applied on the training data matrix $\mathbf{F} = \{\mathbf{f}_1, \mathbf{f}_2, ..., \mathbf{f}_M\}$ (with proper alignment), which consists of the coordinates of landmarks of all the *M* training images. Then each shape in the training set can be approximated using the mean shape and a weighted sum of the first *t* largest eigen vectors:

$$\mathbf{f} = \overline{\mathbf{f}} + \Phi \mathbf{b} \tag{2.2}$$

where $\overline{\mathbf{f}}$ is the mean shape and $\boldsymbol{\Phi} = [\varphi_1, \varphi_2, ..., \varphi_t]$ is the matrix of the first *t* eigen vectors and **b**=[b_1, b_2, ..., b_t]^T is given by $\mathbf{b} = \boldsymbol{\Phi}^T(\mathbf{f} - \overline{\mathbf{f}})$. Vector **b** defines the set of parameters of a deformable model and Eq. (2.2) allows us to generate new shapes by varying the elements of **b** within suitable limits. In such modeling, all possible deformations are based on the variations in the training data and are jointly controlled by **b** without intuitive correspondence between the parameters and specific shape deformations.

In the proposed BASM, we define a key point for each of the seven point-paths shown in Figure 4. Instead of applying PCA on the training data matrix directly, we first normalize the face shape by scaling the eye distances to a fixed value, and then align each point-path by moving the key points to the prior positions that are determined from anthropometric face measurements (normalized to the fixed eyes distance). Then each face shape can be represented as

$$\mathbf{f} = \gamma(\mathbf{f}' + r(\mathbf{l})) \tag{2.3}$$

in which γ is the scaling factor, **l** is the offsets of path alignments, $r(\mathbf{l})$ is a mapping from 7 key points to 58 anchor points by duplicating the offset of each key point for all the anchor points in the corresponding point-path, and **f**' is the shape after the adjustments.

Further, we form matrix $\mathbf{F}' = [\mathbf{f}_1', \mathbf{f}_2', ..., \mathbf{f}_M']^T$ for all adjusted training shapes and perform

PCA on this matrix. Now each original face shape can be further formulated by

$$\mathbf{f} = \gamma(\overline{\mathbf{f}}' + \Phi'\mathbf{b}' + r(\mathbf{l})) \tag{2.4}$$

where $\overline{\mathbf{f}}' = \frac{1}{M}\sum_{i=1}^{M}\mathbf{f}_i'$ is the mean of $\mathbf{F}'$, $\mathbf{\Phi}'$and $\mathbf{b}'$ are the eigen vector matrix and eigen values respectively. Note that, $\mathbf{\Phi}'$and $\mathbf{b}'$ are obtained from $\mathbf{F}'$ instead of the original $\mathbf{F}$, thus they are only responsible for *local* shape variations of each component (the corresponding key point of the component is fixed), which is different from that in ASM. For convenience, we use $\mathbf{\Phi}$ and $\mathbf{b}$ for $\mathbf{\Phi}'$and $\mathbf{b}'$ in the rest of this section.

In order to further generalize this deformable model, we introduce two more parameters: $\mathbf{s}$ to control the aspect ratio of the key point set, and α to control the horizontal off-plane rotation. This gives us:

$$\mathbf{f} = \gamma(\overline{\mathbf{f}}' + \Phi\mathbf{b} + r(\mathbf{l}) \otimes \mathbf{s}) \odot \alpha \tag{2.5}$$

where $\odot$ indicates horizontal off-plane rotation and $\otimes$ denotes the net effect the parameter $\mathbf{s}$ has on $r(\mathbf{l})$.

With Eq. (2.5), we can generate new shapes by varying the parameter $\boldsymbol{\theta} = \{\mathbf{b}, \mathbf{l}, \mathbf{s}, \gamma, \alpha\}$. Figure 5 illustrates the respective effects of the parameters on shape deformations. Intuitively, $\mathbf{b}$ is for deforming all the component shapes with their corresponding key points fixed, $\mathbf{l}$ is computed from prior knowledge of human face shape and is used for adjusting the positions of each face component, $\mathbf{s}$ corresponds to the aspect ratio of the face region, γ controls the scale of the shape, and α controls horizontal off-plane 3D rotation. Specifically, given a 2D shape $\mathbf{g}$, horizontal off-plane 3D rotation of angle α is defined as

$$\mathbf{g} \odot \alpha = [\mathbf{G} - r(C_G)] \cdot \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 1 & 1 & 1 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix} + r(C_G) \tag{2.6}$$

in which $\mathbf{G}$ is the 3D version of $\mathbf{g}$ with additional depth $\mathbf{d}, \mathbf{G} = [\,\mathbf{g}\ \ \mathbf{d}\,]$, C_G is the rotation center, and r is a 1-to-58 duplication mapping of the 3D coordinates. For simplicity, we assume a single face depth model that was obtained from averaging 330 3D face scans from 111 different people [25]. The rotation center is assumed to be located at half of the face width behind the tip of the nose.

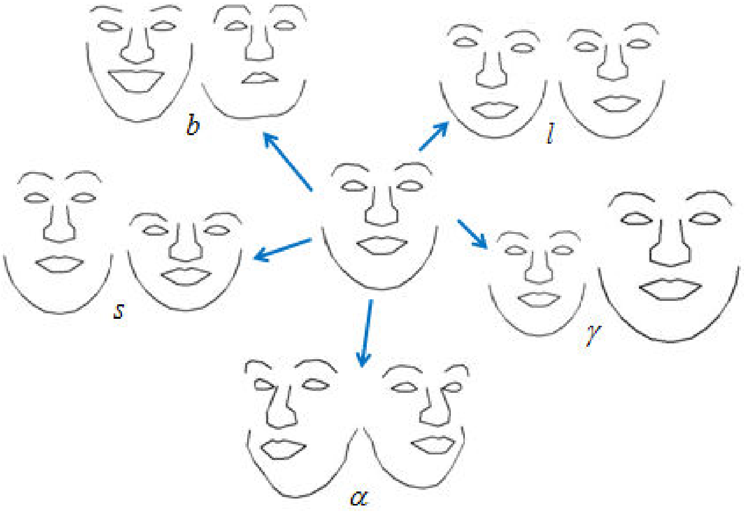


Figure 5: Effects of varying different parameters in BASM.

Compared with Eq. (2.2), Eq. (2.5) incorporates prior shape information by introducing parameter **l** (the offsets of each point-path from the positions of prior key points). Furthermore, Eq. (2.5) explicitly employs separate parameters for different deformations, making it possible for imposing appropriate constraints on each parameter so as to generate more physically-meaningful shapes. Prior key points are computed based on the face position (obtained from a face detection step) and anthropometric face constraints. More will be discussed in Section 2.4.

2.3.2 Bayesian Parameter Update with Statistical Sampling

For a given face image, the generic face model obtained above needs to be updated to best match to the input. This requires the update of the model parameter $\boldsymbol{\theta} = \{\mathbf{b}, \mathbf{l}, \mathbf{s}, \gamma, \alpha\}$ given the input image as observation. We formulate this process as a Bayesian estimation problem, i.e., the estimation of the posterior density of $\boldsymbol{\theta}$ given an input image I:

$$p(\boldsymbol{\theta} \mid I) = \frac{p(I \mid \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(I)} \propto p(I \mid \boldsymbol{\theta}) p(\boldsymbol{\theta}) \tag{2.7}$$

```
Initialize $\boldsymbol{\theta} = \{\mathbf{b}, \mathbf{l}, \mathbf{s}, \gamma, \alpha\}$ as $\boldsymbol{\theta}_0 = (\mathbf{b}_0, \mathbf{l}_0, \mathbf{s}_0, \gamma_0, \alpha_0)$
$\mathbf{b}_0 = zeros(1,t)$, $\mathbf{l}_0 = zeros(7,2)$, $\mathbf{s}_0 = (1,1)$, $\gamma_0 = 1, \alpha_0 = 0$
(1) Generate $L$ random samples $\boldsymbol{\theta}_i, i = 1,...,L$;
(2) Loop until a fixed number of iterations completed
        (2.1) Compute the likelihood for each sample;
        (2.2) Re-sample proportional to the likelihoods.
(3) Compute the final model from the weighted samples.
```

Figure 6: Proposed statistical sampling algorithm.

In general, the above density is multi-modal and the dependency model on I will be highly nonlinear. Thus using a parametric form for the density would be challenging. Consequently, we propose to use a statistical-sampling-based algorithm for the estimation problem, as done in [33]. Essentially, $p(\boldsymbol{\theta} \mid I)$ is approximated by a set of samples of $\boldsymbol{\theta}$ with proper weights. At the beginning, the samples are drawn around the parameters initialized by face detection and a generic face model. The samples will then be updated iteratively based on the given image. This leads to a particle filtering scheme as summarized in Figure 6.

Proper constraints need to be imposed in generating random samples in the parameter space since many samples correspond only to implausible configurations (i.e., invalid facial structures). A hyper-rectangle based on the eigen values of the training data matrix or more complex constraints can be set for **b** in ASM (e.g., as done in [33]). However, the nature of ASM (in which all variations are jointly controlled by **b**) prevents accurate constraints from being applied for generating physically-meaningful shapes. In the proposed BASM, parameters are separated for different deformations, thus it provides a more natural framework for imposing appropriate constraints. These are defined in the below:

b: Uniform distribution within the hyper-rectangle as defined in [33], which is learnt from the training data. (The reason we choose uniform distribution is that **b** is for controlling detailed shape variations which are mostly related to appearance and expression. It is not meaningful to assume any specific distribution on a limited number of face images obtained from random subjects.)

l and **s**: Gaussian distributions for **s** and each element in **l**, with means and variances obtained from anthropometric face priors. Details will be discussed in Section 2.4

γ : Rough face scale can be estimated by the bounding box of the detected face region. The range of uniform random sampling γ is set according to the assumed performance of face detector to compensate for its inaccuracy.

α : Uniform distribution within the allowed rotation range. We use $[-40^\circ, 40^\circ]$ in our experiments.

The likelihood of each generated sample will be computed and used to update the weight of this sample. This is largely based on the comparison of the gradient patterns in the input image and the learnt profiles of the training images. (Details of Step 2.1 are to be presented in Section 2.5.) The iterative algorithm terminates when reaching the predefined number of iterations.

2.3.3 Semantic-aware Model Enrichment

Although aligned facial landmarks and the connected paths are able to retain the major shape contours of a human face, they are still too simplistic for final tactile representation. With obtained semantic information of the face (i.e., the positions of the face components), we employ a set of processing steps to further enrich the components of the sketchy model depending on their respective semantics. For example, we enrich the model by adding more details, using edge segments from edge detection. We also use the strength of the gradient on a major contour to modulate the tactile pattern (e.g., line width) in rendering this contour. Further, Braille annotations can be added to facilitate understanding. More details on these strategies will be described in Section 2.6.

2.4. Anthropometric Face Priors

As discussed in Section 2.3.1, one essential difference between the proposed BASM and ASM is the incorporation of anthropometric prior information into the model. The book Anthropometry of the head and face by Farkas [36] describes elegant methods and results for measuring human head and face based on thousands of human subjects. The measurements of human face and head are approximately described by Gaussian distributions with means and standard deviations. In our work, we adopt model priors including reference key point set (used for computing l in Eq. (2.5)) and the sampling constraints for *s* and *l* (Section 2.3.2) from anthropometric statistics provided by this book. In the below, we first describe the chosen landmarks and distances in Section 2.4.1, and then the computation of the priors for BASM in Section 2.4.2.

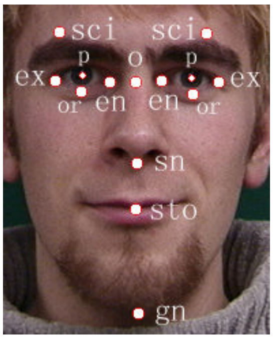


Figure 7: Anthropometric landmarks.

2.4.1 Anthropometric Face Measures

From [36], we select 13 landmarks and 7 distances which are relevant to the 7 point-paths in the 58-anchor-point model. We add a virtual landmark ***o*** as a reference point. Figure 7 illustrates all the landmarks. The 7 distances adopted from [36] include ***en_en***↔, ***ex_ex***

↔, ***sci_or***↕, ***p_or***↕, ***en_gn***↕***,*** ***sto_gn***↕, ***sn_gn***↕. ("↔" and "↕" denote horizontal and vertical distances respectively.)

2.4.2 Anthropometric Face Priors for BASM

Based on the statistical data from [36], we first compute 4 ratios that are scale and position invariant. These ratios are computed based on the distances D_0, D_1, $D_{2,}$ $D_{3,}$ D_4 as defined below and illustrated in Figure 8. The distance D_0 between the two eyes is used as a reference, and all the ratios are computed with respect to $D_{0.}$

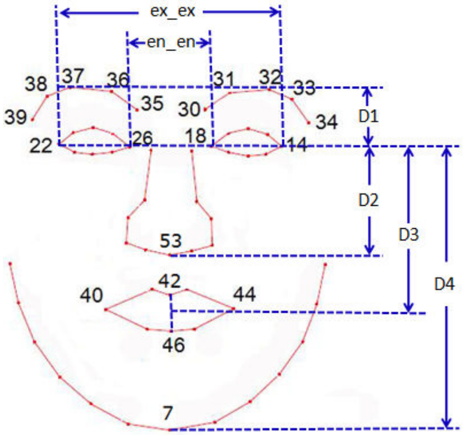


Figure 8: Anthropometric distances.

$$D_0 = \frac{(en\_en + ex\_ex)}{2}$$

$$D_1 = sci\_en \approx sci\_p$$

$$= sci\_or - p\_or$$

$$D_4 = en\_gn$$

$$D_3 = D_4 - sto\_gn$$

$$D_2 = D_4 - sn\_gn$$

The ratios D_1/D_0, D_2/D_0, D_3/D_0 and D_4/D_0 are recorded as anthropometric constraints.

With these constraints, we can further compute the prior key points of face components (Section 2.3.1) and parameter constraints for sampling (Section 2.3.2) as shown in the below.

Prior Key Points ${}^{k}B$: Given the pre-calculated anthropometric face constraints, ${}^{k}B$ can be computed by using the following equations:

Basis: $o_h = \dfrac{p_{18h} + p_{26h}}{2}$, $o_v = \dfrac{p_{18v} + p_{26v}}{2}$

Chin/Jaw: ${}^{k1}B_h = o_h$, ${}^{k1}B_v = o_v + d_0 \times \dfrac{D_4}{D_0}$

Left eye: ${}^{k2}B_h = p_{18h}$, ${}^{k2}B_v = p_{18v}$

Right eye: ${}^{k3}B_h = p_{26h}$, ${}^{k3}B_v = p_{26v}$

Left eyebrow: ${}^{k4}B_h = \dfrac{p_{34h} + p_{30h}}{2}$,

$${}^{k4}B_v = p_{18v} - d_0 \times \frac{D_1}{D_0}$$

Right eyebrow: ${}^{k5}B_h = \dfrac{p_{35h} + p_{39h}}{2}$,

$${}^{k5}B_v = p_{26v} - d_0 \times \frac{D_1}{D_0}$$

Mouth: ${}^{k6}B_h = o_h$, ${}^{k6}B_v = o_v + d_0 \times \dfrac{D_3}{D_0}$

Nose: ${}^{k7}B_h = o_h$, ${}^{k7}B_v = o_v + d_0 \times \dfrac{D_2}{D_0}$

In the training stage, average eye distance $\bar{d}_0$ and average positions of all training data $\bar{p}_{18}$, $\bar{p}_{26}$, $\bar{p}_{30}$, $\bar{p}_{34}$, $\bar{p}_{35}$ and $\bar{p}_{39}$ are used as inputs. In the test stage, p_{18}, p_{26}, p_{30}, p_{34}, p_{35} and p_{39} are also set as the mean values of the training data by approximation in our experiments.

Prior Distributions for Random Sampling: For sampling **s** and **l** in Eq. (2.5), we use Gaussian distributions (illustrated in Figure 9 for **l**) with variances calculated by combining the standard deviations of the anthropometric distance measurements used in computing D_1~D_4.

$$\mathbf{s} \sim G([0,0],[\sigma_h^2,0;0,\sigma_v^2]) \tag{2.8}$$

$${}^{i}\mathbf{l} \sim G([{}^{k_i}B_h, {}^{k_i}B_v],[{}^{i}\sigma_h^2,0;0,{}^{i}\sigma_v^2]),\ i \in \{1,4,5,6,7\} \tag{2.9}$$

The distributions defined in Eq. (2.9) are for face components except the eyes. To compensate for the inaccurate eye positions (since we used an approximation discussed earlier), we define the following Gaussian distributions to capture the possible inaccuracy, leading to probabilistic prior key points for the eyes,

$${}^{i}\mathbf{l} \sim G([0,0],[{}^{i}\sigma_h^2,0;0,{}^{i}\sigma_v^2]), i \in \{2,3\} \tag{2.10}$$

(Since we have normalized the eye distance and aligned the eyes to fixed positions, the referred distributions are of zero mean.) In this case, the random parameter sampling step consists of two stages: sample prior eye positions according to the distributions in Eq. (2.10) first and then sample the aspect ratio **s** and other prior path positions by using distributions in Eq. (2.8) and Eq. (2.9) respectively.

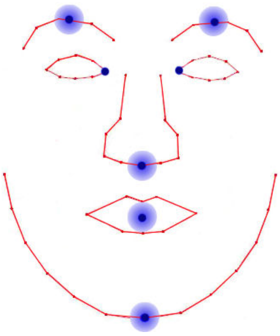


Figure 9: Key point and their prior distributions.

2.5. Computing the Shape Likelihood

The likelihood computation for a sample in Step 2.1 of the algorithm of Figure 6 is an essential step for the statistical sampling procedure. In this section, we describe the details of estimating the likelihood of the generated random samples, $p(I\,|\,\boldsymbol{\theta})$. We use both local gradient profile [19] and edge in this evaluation. Let g_j denote the gradient pattern computed along the line perpendicular to the boundary of a shape instance $\mathbf{f}_i$ through landmark point p_j. And E_I represents the edge map of image *I*. We define the likelihood as

$$p(I \mid \boldsymbol{\theta}_i) = \eta \cdot p(g_1, \cdots, g_N \mid \boldsymbol{\theta}_i) + (1-\eta) \cdot p(E_I \mid \boldsymbol{\theta}_i) = \eta \cdot \left(\prod_{j=1}^{N} p(g_j \mid \boldsymbol{\theta}_i) \right) + (1-\eta) \cdot p(E_I \mid \boldsymbol{\theta}_i) \tag{2.11}$$

The likelihood consists of two terms: model-driven term $p(g_1, \cdots, g_N \mid \boldsymbol{\theta}_i)$, which is the joint probability of the gradient profiles at the N local landmarks given the shape configuration $\boldsymbol{\theta}_i$; and data-driven term $p(E_I \mid \boldsymbol{\theta}_i)$, which measures how well a generated face shape matches the detected edges on the face. These will be discussed in more detail in the below after the introduction of an illumination invariant feature in Section 2.5.1.

2.5.1 Illumination Invariant Feature

To ensure the computed gradient profiles to be more or less invariant to illumination, we first preprocess the image by adopting the method from [21]: the image is first divided into patches and then normalized with respect to local illumination conditions, which are approximated by a low-pass version of the local patch, as shown in Eq. (2.12)

$$R = \frac{O}{O * F_l} \tag{2.12}$$

where *O* is the original image patch, F_l is a low pass filter and *R* is the image patch after this “normalization”. A smoothing step follows to eliminate the “blocky artifacts” (We use Gaussian low-pass filter for smoothing in our experiments). Figure 10 shows two examples, where the images on the right would lead to more balanced gradient computation for both sides of the faces.

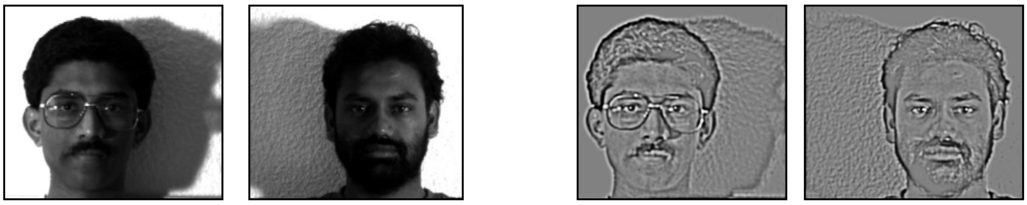


Figure 10: Two examples of illumination invariant feature.

2.5.2 Pose-dependent Local Gradient Profile

The local appearance models, which describe local image features around each landmark, are modeled as the first derivative of the intensity pattern, g_j, computed along the line perpendicular to the boundary of a shape instance $\mathbf{f}_i$ through landmark point p_j. As illustrated in Figure 11, for landmarks on the chin path, only patterns on the inward side are considered (*length* = 15 pixels in our experiments); for all other landmarks, gradient patterns lie on both sides of the point are extracted (*length* = 31 pixels in our experiments). Note that, in the training set, the gradient profiles at each anchor point vary from image to image, and from pose to pose. For instance, the mean gradient profile of p_1 computed over faces that turn left could dramatically differ from the mean gradient profile of p_1 computed over faces that turn right, as illustrated in Figure 12.

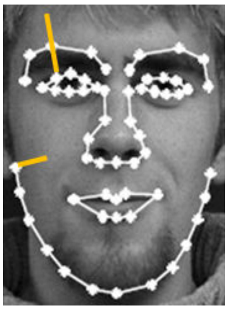
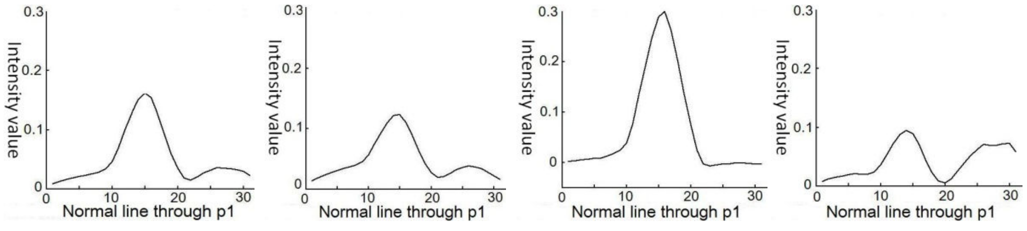
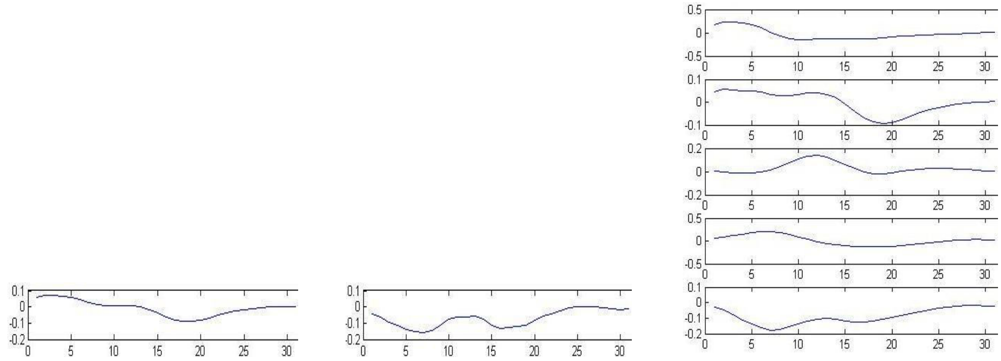


Figure 11: Illustration of local normal lines.



(a) (b) (c) (d)

Figure 12: The mean gradient profile is very sensitive to pose. Mean gradient profile at p_1 computed over all the training images (a), over the nearly frontal pose set (b), over the turning-right pose set (c) and over the turning-left pose set (d) look quite different from each other. (Right/left is in terms of the face in the image.)



(a) (b) (c)

Figure 13: (a) The mean gradient profile at p_{51} computed over all the training images in subset Γ. (b) The gradient profile at p_{51} for a testing image. (c) The 5 cluster centroids obtained from clustering in subset Γ. The score of matching (b) to (a) would be very low, whereas (b) matches well to the 5th cluster centroid in (c), indicating that k-means clustering can capture more appearance variations in the training set than simply using the mean gradient profile.

Therefore, using the mean profile averaged across all poses, as done in [9], may not give a good template for the corresponding anchor point. To remedy this, we propose a pose-dependent local appearance model. Specifically, we divide the entire training image set into three subsets based on the pose variations:

$$I(\alpha) \in \begin{cases} \Gamma : -15^\circ \le \alpha \le 15^\circ, \text{frontal} \\ \Theta : \alpha > 15^\circ, \text{turn left} \\ \Lambda : \alpha < -15^\circ, \text{turn right} \end{cases} \qquad (2.13)$$

For each landmark p_j, we first apply k-means clustering (with 5 clusters) to all the gradient profiles of p_j in set Γ, Θ and Λ respectively, and then record the 5 cluster centroids for each set as the templates, denoted as $\bar{g}_{vj}(I(\alpha)), v = 1,\ldots,5$ for p_j. Figure 13 illustrates that this scheme with 5 templates for each pose can better capture appearance variations in the training set than simply using the mean gradient profile.

2.5.3 Matching Using Weighted Bhattacharyya Distance

Given a configuration $\boldsymbol{\theta}_i = \{\boldsymbol{b}_i, \mathbf{l}_i, \mathbf{s}_i, \gamma_i, \alpha_i\}$, the shape will be compared to its corresponding training set dependent on the value of α_i (see Eq. (2.13)). And the model-driven likelihood term is defined as

$$p(g_1, g_2, ..., g_N \mid \boldsymbol{\theta}_i) = \exp(-\sum_{j=1,...,N} D(g_j, \overline{g}_j)) \tag{2.14}$$

in which, N is the number of landmarks and $D(g_j, \overline{g}_j)$ is the distance between the sample gradient pattern for landmark p_j and the average gradient pattern among training data for p_j. If pose specified by Eq. (2.13) is taken into consideration, Eg. (13) can be rewritten as

$$p(g_1, g_2, ..., g_N \mid \boldsymbol{\theta}_i) = \exp(-\sum_{j=1,...N} D(g_j, \overline{g}_j(I(\alpha_i)))) \tag{2.15}$$

To define the distance metric $D(\mathbf{p},\mathbf{q})$ between two gradient patterns $\mathbf{p}$ and $\mathbf{q}$, we use the Bhattacharyya distance [37]

$$D(\mathbf{p},\mathbf{q}) = -\ln(\sum_{x \in X} \sqrt{p_x \cdot q_x}) \tag{2.16}$$

which was found empirically to be better than simple Euclidean distance in our experiments.

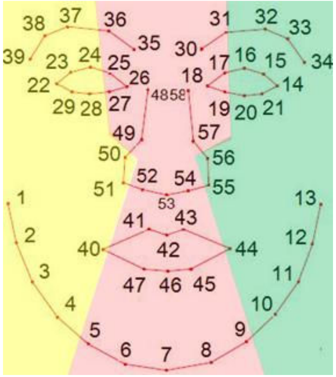


Figure 14: Partitions of landmarks: yellow-S_1, red-S_2 and green-S_3.

For a hypothesized face shape with horizontal off-plane rotation angle α, the algorithm chooses the corresponding template set to compute a matching score at testing landmark p_j as the minimum distance between g_j and one of the 5 centroids:

$$D(g_j, \overline{g}_j(I(\alpha))) = \min_{v=1,\ldots,5}\{D(g_j, \overline{g}_{vj}(I(\alpha)))\} \tag{2.17}$$

We also observed that when the face turns left/right, the collection of landmarks located on the left/right side of the face would have smaller contribution to the likelihood computation. Given the rough orientation of a shape instance, the likelihood model can be further improved such that the image measurements of different set of landmarks are weighted differently. To implement this idea, we partition the *N* landmarks into three sets, S_1, S_2 and S_3, corresponding to the set of right-side, middle and left-side landmarks (see Figure 14). If a sample model parameter $\boldsymbol{\theta}_i$ tells that the face turns to the right, the landmarks in set S_1 will be assigned smaller weights than those in S_2 and S_3 in computing the distance of Eq. (2.16); similar for other poses. This yields the following enhanced likelihood model:

$$p(g_1, g_2, \ldots, g_N \mid \boldsymbol{\theta}_i) = \exp(-\begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix}^T \cdot \begin{pmatrix} \sum_{j\in S_1} \min_{v=1,\ldots,5}\{D(g_j, \overline{g}_{vj}(I(\alpha_i)))\} \\ \sum_{j\in S_2} \min_{v=1,\ldots,5}\{D(g_j, \overline{g}_{vj}(I(\alpha_i)))\} \\ \sum_{j\in S_3} \min_{v=1,\ldots,5}\{D(g_j, \overline{g}_{vj}(I(\alpha_i)))\} \end{pmatrix}) \tag{2.18}$$

where $\boldsymbol{\theta}_i = \{\boldsymbol{b}_i, \mathbf{l}_i, \mathbf{s}_i, \gamma_i, \alpha_i\}$ and $[c_1, c_2, c_3]^T$ contains the weights for the three landmark sets respectively.

2.5.4 Data-driven Likelihood Term

As mentioned in the beginning of this section, the data-driven term in likelihood estimation measures how well a face shape sample fits to the detected edges on the face. This is formally achieved by computing the term as

$$p(E_I \mid \boldsymbol{\theta}_i) = \frac{\beta_i}{\beta} \tag{2.19}$$

In Eq. (2.19), β_i denotes the total number of edge pixels encompassed by all the 3×3 windows centered at each point located on the facial feature contour that fall inside of the bounding box of the true face obtained by face detection; β denotes the total number of edge pixels that fall inside of the bounding box of the true face obtained by face detection. Such $p(E_I | \boldsymbol{\theta}_i)$ gives us the ratio of the model edge pixels to the true face edge pixels. As illustrated in Figure 15, the face shape in the left image matches the edge map better than that in the right one, thus ideally, $p(E_I | \boldsymbol{\theta}_1) > p(E_I | \boldsymbol{\theta}_2)$.

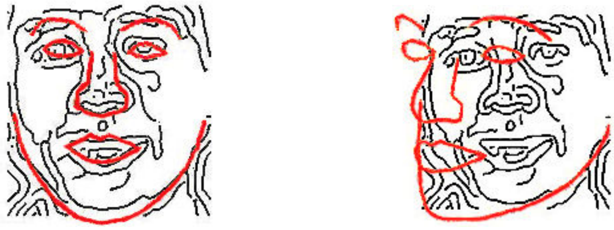


Figure 15: Illustrations of data-driven part: Face shapes in left and right images are specified by configuration $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ respectively (Two edge maps are the same).

2.6. Semantic-aware Processing for Multi-model Tactile Rendering

To evaluate the effectiveness of the proposed approach, we performed both objective evaluation on the proposed BASM-based face alignment algorithm (Section 2.7.1) and subjective evaluation on the usefulness of the tactile faces produced by the approach (Section 2.7.2).

2.6.1 Component-specific Enrichment

In tactile representation, in addition to the contours of the major facial features like eyes, mouth, and nose, it is also critical to keep other informative edges in final tactile rendering. To achieve this goal, we enrich the basic face model obtained from the face alignment algorithm with the edges detected by a Canny edge detector. An adaptive edge refining step is used to filter out the redundant details. This process is semantic-aware in the sense that the refining step varies depending on where on the face the algorithm is applied to. Specifically, we perform the following processing:

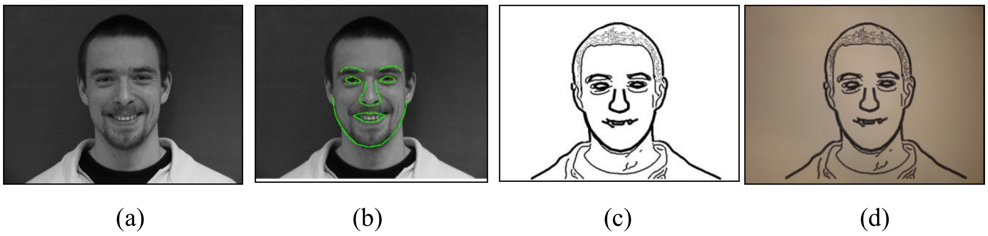
- Eyes, eyebrows, jaw and nose: We keep their shape contours and the nearby edge segments to render some details such as wrinkles in the original image.

- Mouth: We keep the edges in the mouth region which is defined as an enlarged bounding box centered at the aligned mouth contour.
- Chin and face region: We define the face region by mirroring the aligned contour of the chin up and use the projected curve as the upper bound of the face region. In order to make the major face components salient for tactile sensing, edge details within this region (except those retained by other steps) are cleaned up.
- Hair, ears, neck and shoulders: We estimate the outer region of the portrait based on the aligned face shape and retain the edge/texture details if these components exist in the image.

2.6.2 Other Enhancements

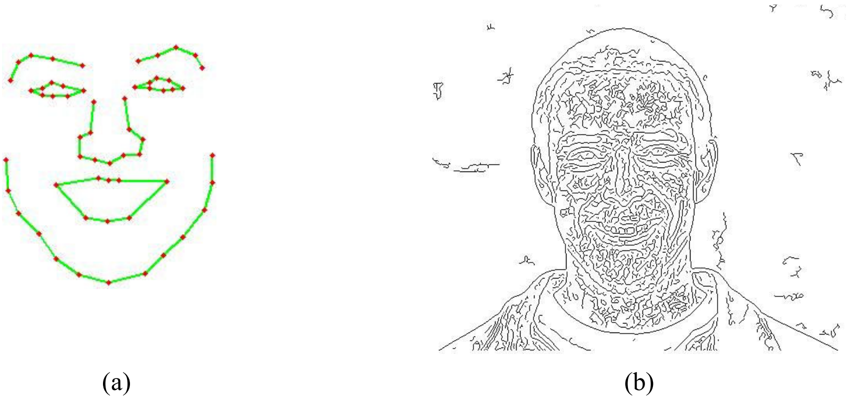
The edge-enriched face sketch can be transformed to tactile form by printing it out through a tactile embosser or a thermal enhancer. In this stage, we exploit the strength of the gradient to modulate the tactile patterns in generating the tactile graphics. For example, we use denser dot patterns for areas with strong gradients, thicker lines for major facial features, and thinner lines for the secondary features including wrinkles, fine edges around the eyes and the mouth.

It is also possible to insert Braille annotations to the final tactile graphics to further assist the blind user in comprehending the tactile printout. These annotations may come from the face alignment step (e.g., Braille text “nose” placed close to the aligned nose contour), or may even be extracted from the metadata of the underlying image. These types of annotations, if combined with a multimodal system such as an interactive tactile touchpad (e.g., IVEO touchpad [38], may convey more information than the tactile lines alone).



(a) (b) (c) (d)

Figure 16: The visual-tactile conversion process: (a) Original image; (b) Result of face alignment; (c) Edge-enriched and gradient enhanced representation; (d) Tactile printout from a thermal enhancer.



(a) (b)

Figure 17: Results without semantic-aware processing: (a) Using only the matched face shape of Figure 16-(a); (b) Using the edge map of Figure 16-(a) generated from Canny edge detector with default parameters.

Figure 16 illustrates the results of the tactile conversion process with an example, in which (a) is the input image, (b) the face alignment result overlaid on the original image, (c) the edge-enriched and gradient enhanced face shape, and (d) the final actual tactile printout from a thermal enhancer.

Figure 17 illustrates the results of using two alternative ways for generating tactile faces without the semantic-aware processing technique. Obviously, simple sketch of a face without hair, neck, shoulder, etc. (Figure 17-(a)) is not a desired representation of the human face in Figure 16-(a). And generating a tactile face by using simple edge detection without high-level guidance (e.g. how to set the parameters for edge detector) is not able to produce a desired result as well. Figure 17-(b) is an edge map of Figure 16-(a) generated from Canny edge detector with default parameters, in which the face region is completely messed up by small line segments. In addition, due to the binary property of swell-paper (either flat or raised which is thermal sensitive), any attempt to generate a tactile face directly from a gray-scale image would typically fail, since the entire face region which is non-white would be raised and all the face components would be unrecognizable.

2.7. Experiments and Evaluations

To evaluate the effectiveness of the proposed approach, we performed both objective evaluation on the proposed BASM-based face alignment algorithm (Section 2.7.1) and subjective evaluation on the usefulness of the tactile faces produced by the approach (Section 2.7.2).

2.7.1 Objective Evaluation on Face Alignment

We used four face image databases to evaluate the performance of the BASM face alignment algorithm. The first database, IMM [34], comprises 240 images from 40 different subjects. We used 200 images from all subjects (5 images of different scenarios from each subject). The different scenarios are listed in the below and sample images with our alignment results are presented in Figure 18: (1) Full frontal, neutral expression, diffuse light; (2) Full frontal, "happy" expression, diffuse light; (3) Rotated approximately 30 degrees to the right, neutral expression, diffuse light; (4) Rotated approximately 30 degrees to the left, neutral expression, diffuse light; (5) Full frontal, neutral expression spot light added at the person's left side. (Currently we only handle horizontal off-plane rotations in the current system, thus the 6th scenario of each subject with arbitrary head rotations were not included in our experiments.) Images of first 30 persons were used for training (150 in total); the remaining 50 images for testing. We do not include any off-frontal images in creating the deformable model. In other words, off-frontal shapes generated in the sampling stage are obtained by applying 3D horizontal off-plane rotation with approximated depth and rotation axis on the built frontal model. Off-frontal image of the first 30 persons (60 images, right part of Figure 18 (a)) are only used in extracting the local gradient patterns of each landmark, which are taken into account in calculating pose-dependent shape likelihood. The major reason of using such an approximation is that anthropometric face constraints are not available for off-frontal faces with arbitrary rotation angles. This reveals one limitation of our approach that it is not able to handle off-frontal rotations with large angle.

The second data set, denoted as "AR200" consists of 200 images of 40 randomly selected subjects (20 male and 20 female) under 5 scenarios ("Neutral expression", "Smile", "Anger", "Left light on" and "Right light on") from Section 1 of the AR face database [42].

The third data set, denoted as "FERET100" consists of 100 images of 50 randomly selected subjects (33 male and 17 female) from Color FERET database [43]. Since in Color FERET database, different subjects have different numbers of images, we select two basic cases with suffix "fa" and "fb" in the titles (we skipped faces with left/right rotations since most off-frontal faces in FERET database are of nearly 90-degree rotation angle, which is beyond the scope we aim at in this work). Both cases are of large variations of lighting condition (e.g. with side-way lighting), face size and skin color (e.g. subject with very dark skin). We manually annotated these 300 images to obtain the ground-truth data.

In addition, we also experimented with a 30-person face database (denoted "30-person data set") that was independently captured in our lab with varying lighting conditions and poses. The resolution of the images in this data set is much lower than the first three data sets.

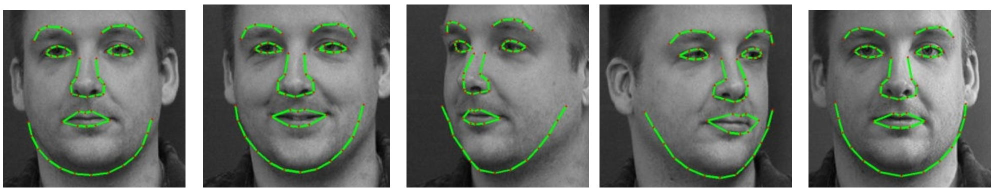


Figure 18: Sample results of images under five scenarios of IMM dataset.

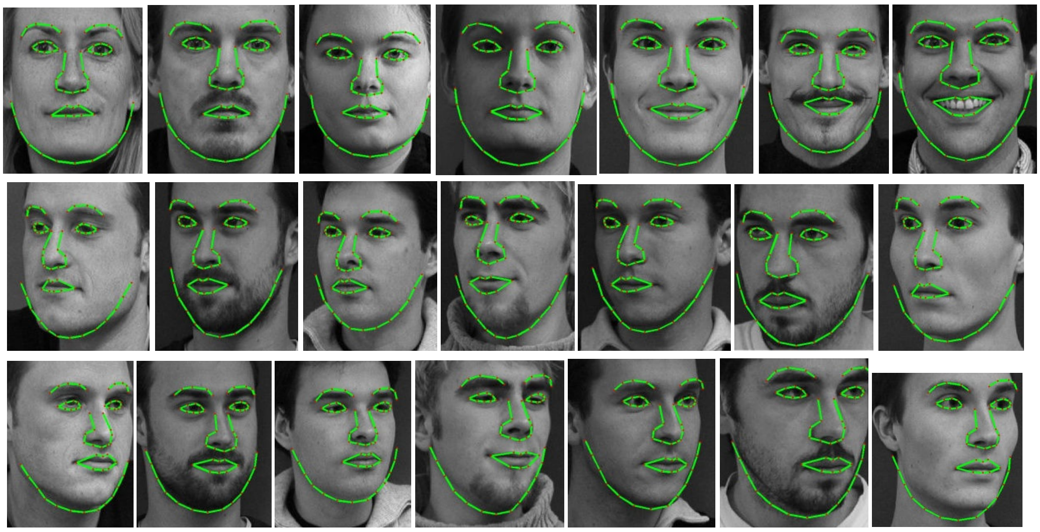
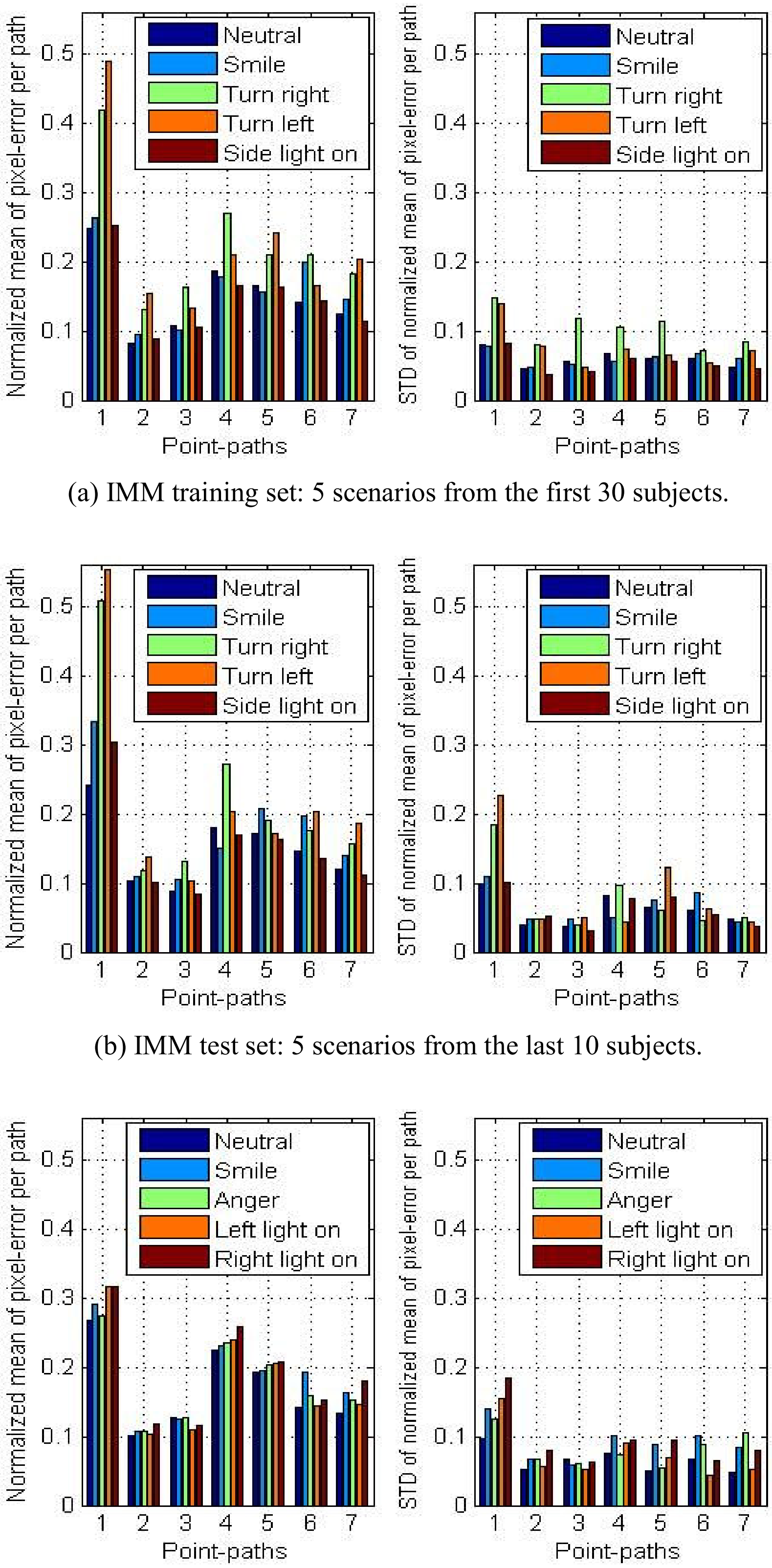


Figure 19: Face alignment results: 1st row-frontal with some variations of expressions and lighting; 2nd row-turn right; 3rd row-turn left.

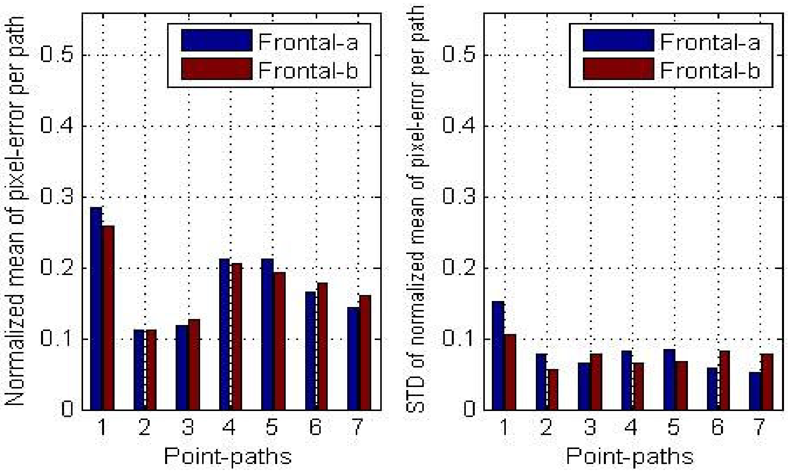
Figure 18 shows the results of all five scenarios of one subject from the IMM test set and Figure 19 shows more results with varied expressions and lighting conditions from both training and testing sets of IMM database. We can see that, the BASM based face alignment algorithm works well with all scenarios of images (Results are slightly better for frontal images than for off-frontal cases, because off-frontal images were not used for creating the deformable model. It is worth pointing out that BASM is able to capture subtle variations of face components due to different expressions (e.g., the mouth regions of the images in the first row of Figure 19), which is helpful for conveying important information in the final tactile representation.



(a) IMM training set: 5 scenarios from the first 30 subjects.

(b) IMM test set: 5 scenarios from the last 10 subjects.

(c) AR200 data set: 5 scenarios from 40 random subjects.



(d) FERET100 data set: 2 frontal sets from 50 random subjects.

Figure 20: Face alignment results on multiple data sets. (Point-paths: 1-chin, 2-left eye, 3-right eye, 4-left eyebrow, 5-right eyebrow, 6-mouth, 7-nose.)

We quantitatively analyzed the performance of the algorithm by computing the average matching errors for the anchor points based on the ground-truth. Figure 20 reports the average errors per anchor points in terms of point-paths (normalized to inner-eye-corner-distance) and corresponding standard deviation over all samples for the above three data sets. (Due to the 58-anchor-point face model, the inner-eye-corner-distance instead of the iris-to-iris distance is used for normalization.) Comparing the accuracies among different point-paths, best performance is achieved on eyes; it is slightly better for eyebrows than for nose and mouth; the worst case occurs on chin. In terms of different scenarios, the algorithm works the best for frontal pose with neutral expression; errors increase when the subject is smiling or one side light is on. Worst cases occur on faces with horizontal rotations. This is not surprising since we did not include any off-frontal images in creating the deformable model. It is worth pointing out that, although the training was done using a subset of the IMM images, the testing results for the AR200 and FERET100 images are equally good, despite the acquisition environments of the databases differ greatly. (The results for the AR200 and FERET100 images are actually slightly better on average but the IMM sets contain more challenging off-frontal images.)

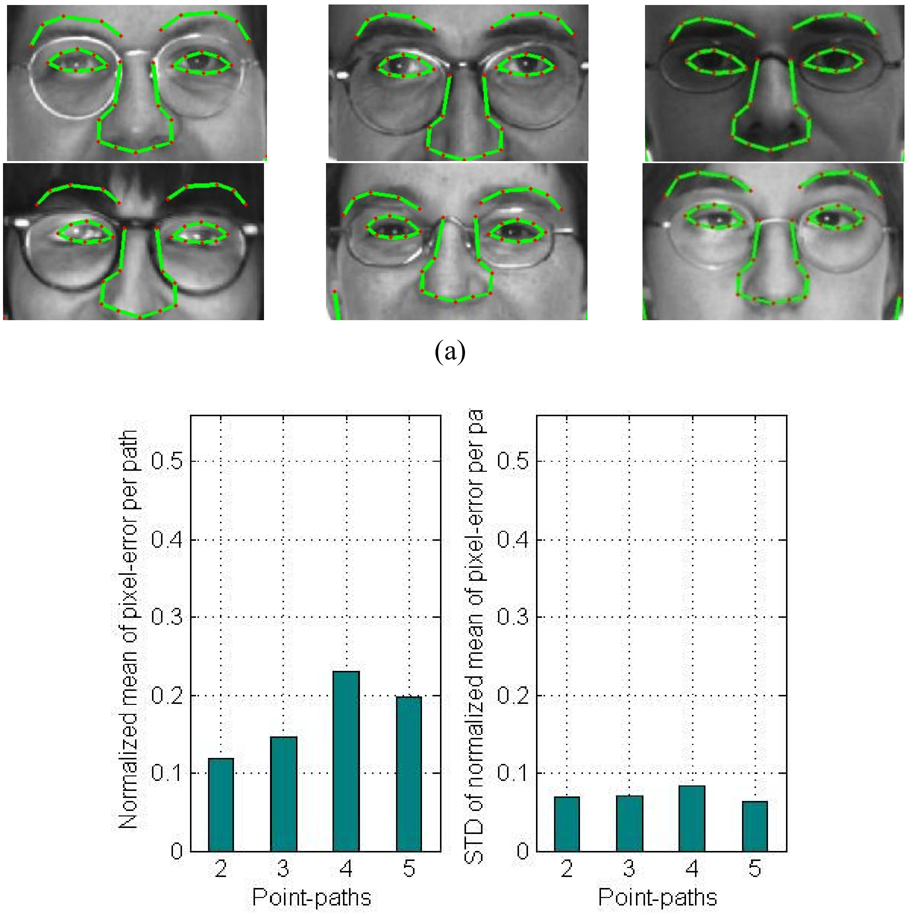
This suggests that our method is very robust with respect to new databases that it never saw in the training stage.

Table 1: Face alignment results of the proposed approach.

Normalized Error per. Anchor Point	≤ 10%	≤ 20%	≤ 30%
IMM Training Set	1.3%	62.7%	93.3%
IMM Test Set	2.0%	54.0%	94.0%
AR200	0.5%	70.0%	97.5%
FERET100	0.0%	72.0%	97.0%

Overall, the average error per anchor point is about 8.6 pixels for the IMM training set, 9.0 pixels for IMM test set, 8.8 pixels for AR200 and 7.8 for FERET100. Since the average height of face regions in the database is about 200 pixels, the error on average is less than 5% of the height of the face region and thus can be deemed as small. The results in terms of the percentage of images with average error per anchor point (normalized to the inner-eye-corner distance) within certain error bounds are summarized in Table 1 (the average inner-eye-corner distances of IMM, AR200 and FERET 100 are 46 pixels, 48 pixels and 51 pixels, respectively). These results improve upon those reported in [33] and are at least comparable to what presented in [21], [26] and [40], although it is difficult to make direct comparison since the landmark model, test images and ground-truth are different. In [32], the best accuracy of errors no-greater-than 8 pixels was reported as 63.9% while the average height of the faces for test is about 180 pixels; Gu et al. achieved an average mis-alignment error 3.49 pixels with all faced normalized to a width of 120 pixels [26]; Liang et. al reported 98.5% and 93.5% cases of errors less than 7.5 pixels on two data sets with the entire images resized to 200-300 pixels [40]. Face sizes in these three papers are much smaller than the test samples we used (e.g. the average face size of IMM database is about 200 190 pixels and it is even slightly larger of AR200 and FERET100), thus the reported errors appear smaller than the results presented in this work. In addition, in our evaluation, we trained the model and the algorithm with a fixed set of images from the IMM database; then the trained model/algorithm was tested on other databases that are completely independent of the IMM database. However, in the experiments of the above three papers, the training and testing sets were formed in such a way that both sets contain images from all the underlying databases. Apparently, our evaluation protocol is much more demanding than that used in the above papers. We attribute the robustness and generalizability of our algorithm with respect to new image databases to the incorporation of the Bayesian prior.

In addition, we also performed separated analysis on subjects who wear glasses or with beards/moustaches. For cases of wearing glasses, we analyzed all 13 subjects (65 images) with glasses in AR200 data set. Sample visual results and quantitative evaluations (including normalized mean errors and corresponding standard deviations over all samples, as shown in Figure 20) of the eyes and the eyebrows are illustrated in Figure 21. Compared to Figure 20-(c), no obvious degeneration happens for point-paths of eyes and eyebrows when subjects are wearing glasses. For subjects with beards and/or moustaches, we analyzed all 14 subjects with light to heavy beards and/or moustaches from IMM database. Figure 22-(a) illustrates two sample results, in which the bottom image is from the subject who might have the heaviest beards among all subjects in IMM database. The result is still reasonably good. More visual results can be found in Figure 19 (i.e. 2nd and 6th column). Figure 22-(b) presents normalized average error plots for point-paths of the chin and the mouth and corresponding standard deviations over all samples. Compared to Figure 20-(a) and (b), accuracies for mouth and chin are comparable to the overall average results.

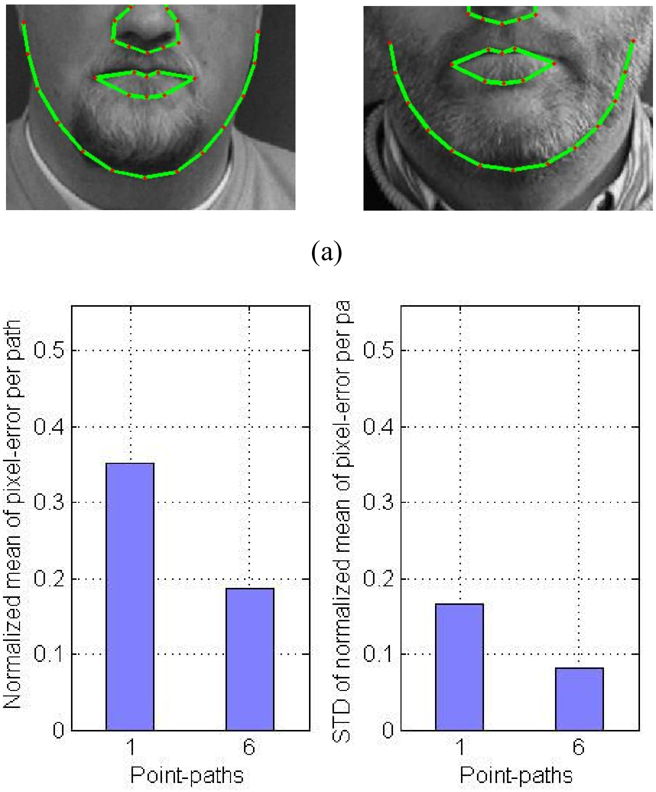


(a)

(b)

Figure 21: AR200-Cases of wearing glasses: (a) Visual results; (b) Quantitative results of eyes and eyebrows over all samples.

Note that the ground-truth shapes labeled manually are not always precise. As illustrated in Figure 23-left, anchor point 48 and 58 (two top landmarks of the nose) are not aligned horizontally. This suggests that the so-called ground-truth is not perfect and thus a relatively large error computed based on the ground-truth needs not mean the matching is poor. For example, the anchor points 5-9 (lower part of the chin) in the two images of Figure 23 can be deemed as perfect fit to the image, but the corresponding anchor points are not exactly at the same positions. These observations explain part of the reasons that we got close-to-zero percentage for cases of “no greater than 0.1 inner-eye-corner-distance”.



(a)

(b)

Figure 22: IMM-Cases with beards and/or moustaches: (a) Sample visual results; (b) Quantitative results of chin and mouth over all samples.

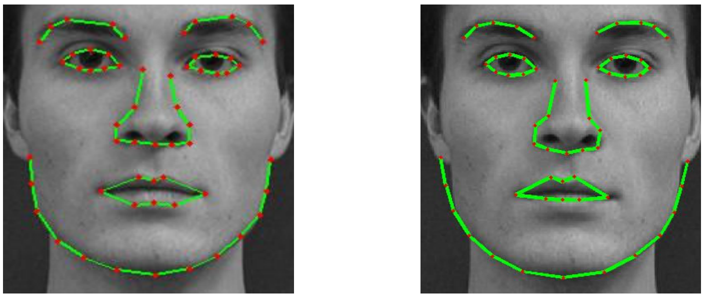


Figure 23: Comparison between the ground-truth (left) and the obtained result (right) with 4.8 pixel error per anchor point.

For the 30-person data set, a few sample results are shown in Figure 24, demonstrating the robust performance of the algorithm. Comparing the 1st row of Figure 24 from our previous approach presented in [33] and the 2nd row from the BASM approach for the same images, obvious improvements can be observed.

In terms of convergence of the iterative sampling process, in our experiments, we observed that it actually converges very fast if the center of the initial face model is placed reasonably close to the ground-truth center (i.e. when the face detection result is reasonably accurate). All reported results in this work were generated with 4 iterations and 200 random samples in each of the iterations.

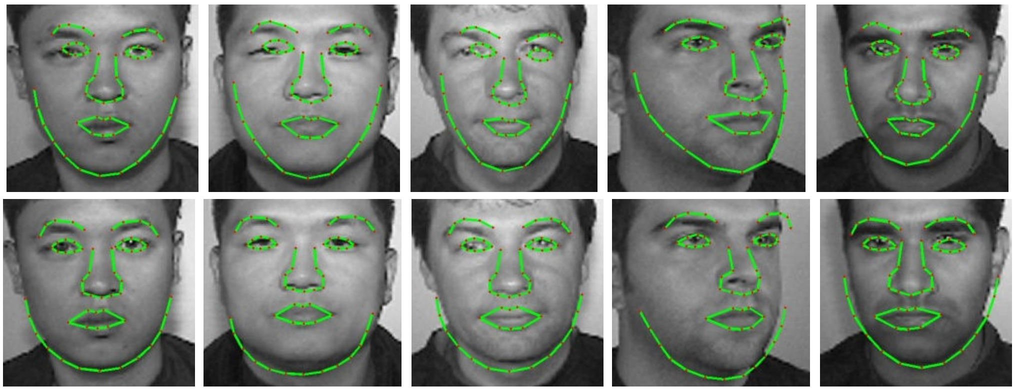


Figure 24: Face alignment results on the 30-person data set: Top row—our previous results presented in [33]; bottom row—BASM results.

2.7.2 Subjective User Evaluation

The ultimate goal of our work is to automatically generate tactile form of face images for visually-impaired people. In Section 2.7.1, we have shown the performance of the BASM based approach on face alignment. In this section, we will present user evaluations of the tactile images created by the proposed approach. The evaluation was done by both blind-folded sighted users and visually-impaired users.

6 visually-impaired users and 11 blind-folded sighted users participated in our experiments. The six visually-impaired users include 5 blind person and 1 low-vision person (who was blind-folded in the experiments). Five of them are Braille users (the other people uses screen reader) and only two of them have a little experience with tactile graphics (limited to simple geometric shapes, such as triangle, square, etc.) Since most of the users do not have any experience with tactile graphics, a short training step was performed before the main evaluation experiment. In the training phase, the users were

given some sample tactile face images of the same format as all the experimental images to explore and to get with the layout of the image and various patterns for different face components. Assistances were provided upon request during the training phase. Blind-folded users are people who did not have any experience with tactile graphics. They were also given the training beforehand. Although the end user of the technology will be people with visual impairment, at this stage of study, to verify that the approach does maintain key "visual" features, it was found that recruiting blind-folded sighted individuals for the evaluation was very helpful since they are able to compare what they feel by touching against what they have seen.

Both groups of users were required to explore some tactile face images generated from our approach and answer the following questions: (1) Can you recognize each face component including the mouth, eyes, eyebrows, nose, chin/jaw? (2) Can you recognize the pose of the person, i.e. is he/she turning left or turning right? (3) Can you recognize the gender of the person? (4) Association: Can you identify two images that represent the same person?

Images in Figure 25 were used for all the questions; Images in Figure 26 were used for association questions only. Table 2 and Table 3 present the resultant statistics of the visually-impaired group and the blind-folded group respectively.

Observations from the above two tables are summarized below:

(1) **Major face components:** They were successfully indentified by most of the users from both groups except the left eyebrow of Figure 25-(e) which is close to some curves of the hair. And in the same image, curves of clothes were misunderstood as chin/jaw by one of visually-impaired users.

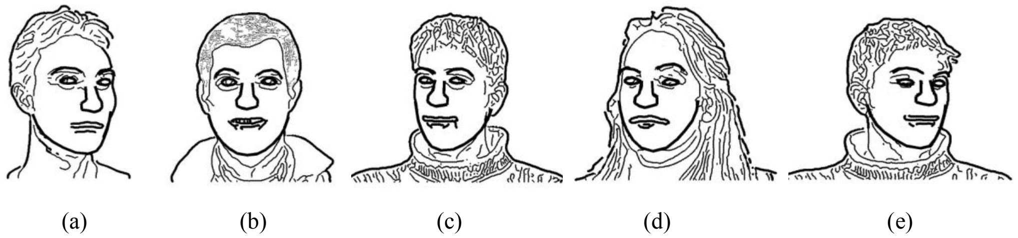


Figure 25: Tactile face images—Set I.

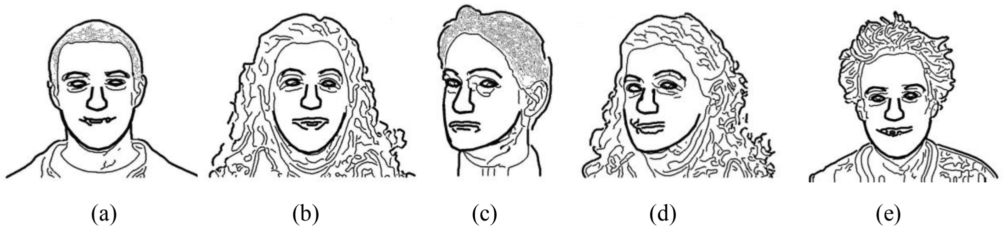


Figure 26: Tactile face images—Set II.

(2) Pose: Both groups achieved high accuracies (only 1 user out of 6 or 11 users at most was wrong for each image).

(3) Gender: This was found to be a tough question. Still, more than 50% of the users in each group got it right for most of the images. The blind/low vision group performed slightly better than the blind-folded group. Length of hair was the main feature used for distinguishing genders by most of the users. Some of them also used sizes of face and eyes for identifications. One of the blind users rejected to use the length of hair as a discriminative criterion, since she has short hair herself. Curves of the shoulder part, which are easily confused with women's long hair, caused most of the mistakes.

(4) Association: Most users found it is the most difficult task, but very interesting on the other hand. Criteria used by different users varied. Most of them relied on properties of hair, such as length, density of hair. Some of them used contours of neck and shoulder regions. In addition, shapes of the chin, mouth and eyes and width of the entire face contributed to some decisions as well.

Table 2: The resultant statistics (number of correct cases) obtained from 6 visually-impaired users.

Image	Left eye, eyebrow	Right eye, eyebrow	Nose	Mouth	Chin/Jaw	Pose	Gender
a	6	6	6	6	6	5	4
b	6	6	6	6	6	5	4
c	6	6	6	6	6	6	3
d	6	6	6	6	6	5	6
e	6	6	6	6	5	6	3
Set	Association						
I	2						
II	5						

Table 3: The resultant statistics (number of correct cases) obtained from 11 blind-folded users.

Image	Left eye, eyebrow	Right eye, eyebrow	Nose	Mouth	Chin/Jaw	Pose	Gender
a	11	11	11	11	10	11	5
b	11	11	11	11	11	10	7
c	11	11	11	11	11	10	4
d	11	11	11	11	11	10	9
e	10	11	11	11	11	11	7
Set	Association						
I	2						
II	2						

(5) Other observations: Most users used eye positions as spatial references for locating other face components. However, it was very interesting that some of the users followed a different way. They started from chin and explored the face bottom-up. We did not observe any salient difference in performance for these two types of approaches. Overall, both groups achieved high accuracies on most of the tasks, while in general, not surprisingly, users from the blind/low vision group were much faster in interpreting the results.

Finally, we report an interesting experiment of human identity recognition based on tactile face images, which was performed with blind-folded participants only. The objective of this experiment is to test whether the proposed approach is able to retain the distinctive characteristics of the facial features. The participants were given two tactile face images generated from our approach and asked to give the identity of each person in the two tactile images respectively by choosing from five names of five persons that they know very well. For example, we asked the participants: "Can you tell who this person is, chosen from Cindy, Troy, Jessie, Michael, and Daniel?" The results were very encouraging: all of the participants were able to correctly recognize the identity of the persons on the two images. This suggests that the automatically created tactile representation indeed retains some distinctive visual features.

2.8. Summary

In this section, we proposed a systematic approach to automatic conversion of facial images into their tactile form. A novel modeling framework, BASM, was proposed, which enables the incorporation of anthropometric priors and facilitates the development of a Bayesian inference algorithm based on statistical sampling. Compared to our recent attempt [33] in addressing this challenging and practical problem, the proposed approach has achieved significant improvement in both accuracy and robustness. Further, comprehensive user evaluation has been reported, based on a group of twelve users including six blind individuals. The results suggest that the proposed approach provides a promising solution to the challenging problem of automatic creation of tactile face images. To our knowledge, this is the first symmetric study on the problem. Generalizing the approach to other types of graphics and building an end-to-end system using the current approach are among our future tasks.

Chapter 3

MINING INTER-MODALITY CORRELATIONS VIA KCCA FOR FLICKR PHOTO TAG RECOMMENDATION

Photo tag recommendation is related to automated image annotation, which has received significant attention in recent years. In many existing approaches, images are divided into sub-regions and a mapping between keywords and sub-regions are learnt. This and similar approaches are useful for images characterized by some key sub-regions but not so effective for generating tags with higher-level semantics that often link to the image as a whole. For example, given an image of the Great Wall, "China" may be one of commonly-used tags, which is unlikely to be directly predicted from purely visual features. To leverage the underlying correlations for compensating the semantic gap, we propose an automatic approach to tag recommendation for a given image without any annotations/tags. Our approach exploits the semantic correlation between image contents and text labels via Kernel Canonical Correlation Analysis (KCCA) [56]. In recommendation, the tags are ranked based on both the image-tag correlation and the input-independent tag popularity learnt from photos with user-created tags.

3.1. Background and Overview of the Proposed Approach

On-line services for archiving and sharing personal photos, such as Yahoo Flickr (www.flickr.com) and Google Picasa (picasa.google.com), have become more and more popular in recent years. Such services effectively provide a virtual social network or community for the users to share their memories, emotions, opinions, cultural experiences, and so on. Interaction among users in such a virtual community is largely enabled by information retrieval and exchange, e.g., photo retrieval and sharing, which are critically facilitated by tags or annotations of the photos. Tag recommendation systems (e.g., [44]) target at assisting users to come up with good tags that are both descriptive for a photo and useful for supporting information retrieval/exchange.

Photo tag recommendation is related to automated image annotation, which has received significant attention in recent years (e.g. [44-48]). For example, Duygulu et al. modeled annotation as machine translation [48], and Mori et al. used co-occurrence models [44]. In both cases, images are divided into sub-regions and a mapping between keywords and sub-regions are learnt. This and similar approaches are useful for images characterized by some key sub-regions but not so effective for generating tags with higher-level semantics that often link to the image as a whole. For example, given an image of the Great Wall, "China" may be one of commonly-used tags, which is unlikely to be directly predicted from purely visual features. Sigurbjornsson and Zwol proposed a tag co-occurrence algorithm for on-line photo tag recommendation [49], where the semantic relationship among tags is used for predicting new tags for a given image with some known tags. The limitation is that, the correlation analysis was purely based on text and thus at least one tag has to be present for the method to work.

In this section, we propose an automatic approach to tag recommendation for a given image without any annotations/tags. Our approach exploits the semantic correlation between image contents and text labels via Kernel Canonical Correlation Analysis (KCCA) [56]. In recommendation, the tags are ranked based on both the image-tag correlation and the input-independent tag popularity learnt from photos with user-created tags. We performed experiments using a realistic database collected on-line to demonstrate the superior performance of the proposed approach. In the following subsections, we first introduce the proposed approach in Section 3.3. Experiments, results and comparisons are presented in Section 3.4 followed by a brief summary of the work in Section 3.5.

3.2. Semantic Image-Tag Correlation Analysis via KCCA

We propose to utilize Kernel Canonical Correlation Analysis (KCCA) to learn the underlying semantic correlation between the visual content and the textual tags of on-line photos and then use the correlation in predicting labels for images without tags. We briefly review the CCA/KCCA algorithm in the below.

CCA attempts to find basis vectors for two sets of variables such that the correlation between the projections of the variables onto these basis vectors is mutually maximized [57]. The canonical correlation between any two data sets is defined as

$$\rho = \max_{W_x, W_y} corr(F_x \cdot W_x, F_y \cdot W_y) \tag{3.1}$$

where F_x and F_y are the two sets of variables, and W_x and W_y are the basis vectors onto which F_x and F_y are projected, respectively. This optimization problem can be formulated as a standard eigen problem [56] which can be easily solved. There may be more than one canonical correlation, each representing orthogonally separate pattern of relationship between the two sets of variables. When extracting the canonical correlation the eigen values are calculated. The square root of the eigen values can be interpreted as the canonical coefficients. Corresponding to each canonical correlation the canonical weights for each of the variable in the data set are calculated. The canonical weights represent the unique positive or negative contribution of each variable to the total correlation.

CCA has been used previously by researchers to find the semantic relationship between two multimodal inputs. In [58], CCA is used to find the language independent semantic representation of a text by using the English text and its French translation as set of variables. In this work, we model the semantic relationship between visual features of an image and the textual annotations/tags through CCA, and use available image features for predicting semantically-related texts in tag recommendation.

CCA is only able to capture linear correlations, while the actual correlation model in our application may be highly nonlinear. Therefore, we resort to the Kernel CCA, which has been used, e.g., in [56] for finding the correlation between image and text features in image retrieval. KCCA projects the data into a higher-dimensional feature space before performing CCA in the new feature space [50]:

$$\phi : x = (x_1,...,x_m) \mapsto \phi(x) = (\phi_1(x),...,\phi_N(x)),\ (m < N) \tag{3.2}$$

where $x = (x_1,...,x_m)$ is a set a variables and ϕ is a mapping from m dimensions to N dimensions. In this work, we use the implementation made available by the authors of [56], with a Gaussian kernel.

3.3. Tag Recommendation: the Proposed Approach

We formulate tag recommendation as a tag ranking problem. We propose a novel approach for ranking all possible tags by weighting image-tag correlation and input-independent tag popularity. An image-tag correlation score is obtained from KCCA and a tag popularity score is defined as the normalized tag frequency in the training set. The weights for the two scores can be used to control the relative contributions from the two parts. Details of the approach are presented in the below.

3.3.1 Tag Ranking

We first collect a training set of photos with corresponding tags. The vocabulary of possible tags is defined as a collection of all tags appearing in the training set: $\{t_j\}, j = 1,...,m$. We define a ranking score for each tag in the vocabulary as below:

$$S_{t_j} = (1-a)\cdot S_{t_j}^{corr} + a \cdot S_{t_j}^{pop} \tag{3.3}$$

where t_j is a possible tag from the vocabulary; $S_{t_j}^{corr}$ and $S_{t_j}^{pop}$ denote the semantic image-tag correlation score and the tag popularity score respectively, and $a \in [0,1]$ is a constant for weighting these two scores.

Compared to existing work this new approach has some advantages. For example, all existing tags from the training set can be recommended. In addition, the weight *a* provides a flexible control of the contributions from the two terms, allowing for example the recommendation to rely more on visual contents or on the tag popularity.

3.3.2 Semantic Image-Tag Correlation Score

We compute the image-tag correlation score S^{corr} based on KCCA as

$$S_{t_j}^{corr} = \max_{i=1\dots n}\{corr_{I_{t_j}^i}\} \tag{3.4}$$

where t_j is a tag from the vocabulary of all possible tags; $I_{t_j}^i$ is the i^{th} training instance (an image with corresponding tags) that t_j belongs to, and $corr_{I_{t_j}}$ is the normalized correlation coefficient between this instance and the input instance. Conceptually, we use the maximum correlation coefficient between the input image and one of the training instances as the semantic image-tag correlation score (correlation coefficients are normalized in Eq. (3.4)). In our experiments, definitions of text and image features are defined in the below.

Text view: In natural language processing (NLP) and text based information retrieval, bag-of-word is one of the most powerful models for describing text [52]. In a bag-of-word model, text is represented as an unordered collection of words disregarding interpunctions and grammar and typically simplified as a Document-Term (DT) matrix, in which each entry of the matrix records the appearance frequency of a specific term in a particular document. To avoid dimension explosion and eliminate noises, terms are carefully selected from a dictionary or from the vocabulary of all available documents using feature selection algorithms.

In this work, we use the bag-of-word model for all text based views in our experiments. Since both of our experiments are category related, we select key words by adopting

TFICF algorithm [60]. Stemming [61] and removing stop [62] words steps are taken before *TFICF.*

$$TFICF(T_k, C_i) = TF(T_k, C_i) \times ICF(T_k) \tag{3.5}$$

$$ICF(T_k) = \log(|C| / CF(T_k)) \tag{3.6}$$

where $TF(T_k, C_i)$ refers to term frequency of term T_k in category C_i; $|C|$ is the number of categories in the collection and $CF(T_k)$ is the category frequency of term T_k. *TFICF* essentially ranks each term by achieving high inner-category frequency and low inter-category existence. Top-k terms are used as key words for each category.

Image view: There are various image features available, serving different purposes in pattern recognition and computer vision problems. For image tag recommendation problem, image features are required to be able to describe both global and local visual information. HSV histograms and image gradient from Gabor filtering are commonly-used local color and texture features of an image. In the meantime, spatial pyramid technique [63] captures the spatial layout information of the image. In this work, we use spatial-pyramid-based HSV histograms and Gabor gradients as the features. Specifically, given an image, we first divide it into blocks. (As shown in Figure 27, we use three layers with 1, 4 and 16 blocks for each layer respectively.) Then, three normalized 8-bin histograms for the HSV channels are computed for each block as the color feature, and 12 gradient energy values from 12 Gabor filters (3 orientations with 4 frequencies) are computed as the texture feature. Gradient energy is calculated by using following equation [45]:

$$|P|^{-1} \sum_{(n,m)\in P} |I_f(n,m)|^2 \tag{3.7}$$

in which, *P* denotes the input image block; I_f is the convolution of image block I and a Gabor filter $G(\theta, f)$ with specified orientation and frequency.

In summary, we have 756-d feature vector for image view, including 504-d $((1+4+16)\times 3\times 8)$ for color and 252-d ($(1+4+16)\times 12$) for texture. Both color features and textures are normalized to $[0,1]$ respectively before further calculations.

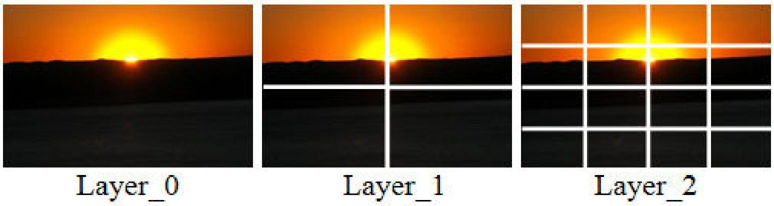


Figure 27: An example of image spatial pyramid.

Assume that we have a training set which contains n instances of images with corresponding tags. The training procedure is described as follows:

1. Extract text and image features from all instances in the training set and form feature matrices $F_x = [f_x^1, ..., f_x^n]^T$ and $F_y = [f_y^1, ..., f_y^n]^T$, where each row represents the feature vector of one instance.
2. Project F_x and F_y to a higher dimensional space by kernel mapping. The projected feature matrices are denoted as F_x*'* and F_y*'*.
3. Perform KCCA between F_x*'* and F_y*'* and find the basis vectors:

$$[W_x, W_y] = KCCA(F_x', F_y') \tag{3.8}$$

   in which W_x and W_y are the found KCCA basis matrices.

4. Project F_x*'* and F_y*'* onto the obtained basis:

$$F_x'' = F_x' \times W_x^k \tag{3.9}$$

$$F_y'' = F_y' \times W_y^k \tag{3.10}$$

   where W_x^k and W_y^k are obtained by selecting top k basis vector from W_x and W_y respectively.

In the test stage, given a new test image, we first extract its image feature f_{y_0} and obtain $f_{y_0}^{'}$ by projecting f_{y_0} to a higher dimensional space using the same kernel mapping as used in Step 2 of the training procedure. We further project $f_{y_0}^{'}$ onto W_y^k as we did for all training images and get $f_{y_0}^{"}$ as a result. Then, the instance level correlation score $corr_{I^i}$ of this image to the training instance i can be computed as a normalized Pearson correlation between $f_{y_0}^{"}$ and $f_{y_i}^{"}$:

$$corr_{I^i} = \frac{correlation(f_{y_0}^{"}, f_{y_i}^{"})}{\max\limits_{i=1...n}(correlation(f_{y_0}^{"}, f_{y_i}^{"}))} \tag{3.11}$$

where $f_{y_i}^{"}$ is the ith row of $F_y^{"}$.

3.3.3 Tag Popularity Score

The tag popularity score is an input-independent score for tags, which describes how likely a word is used as a tag based on the training set. This is defined as:

$$S_{t_j}^{pop} = c_{t_j} / \max_{k=1...m}\{c_{t_k}\} \tag{3.12}$$

where t_j is a tag from the vocabulary of all possible tags and c_{t_j} indicates the counts of appearances of t_j in the training set.

3.4. Experiments and Results

According to a Yahoo study [49], the most frequent classes of tags for Yahoo Flickr photos are *locations*, *artifacts/objects*, *people/groups* and *actions/events*. In our work, we selected two popular topics for each class except *people/groups* (which will be included in future study due to its complexity). Specifically, we picked “office” and “stadium” as *location*, “pyramid” and “Greatwall” for *artifacts/objects*, and “skiing” and “sunset” *for actions/events*. For each topic, we crawled a few thousand images for each topic from Flickr using the FlickrAPI tool. Photos with too few tags (e.g. less than 5) were removed

(For training, images with too few tags may lead to an extreme sparse matrix of text features, which can cause numerical issues to KCCA; for testing, images with too few tags are not appropriate for objective evaluation.) And in order to mitigate user bias (images from the same user are visually very similar in many cases), we keep no more than 15 images from the same Flickr ID. Finally, we used 300 images for each topic, in which 200 images were used for training and 100 images for testing. The training and testing sets were defined by random selections. Obviously, real on-line data is more challenging than research databases (e.g. the databases used in [47]) due to the varying sources of images and the uncontrollable usage of vocabulary in the user-provided tags. In order to show the improvements of the proposed approach, we use the same dataset which was used in our previous work [53].

3.4.1 Evaluation Metrics

Both objective evaluation and subjective evaluation have been performed for validating and assessing the proposed method. For objective evaluation, we compared the recommended tags generated by our approach to the tags provided by the original owners of the photos. If one of user tags is among the recommended tag list, we call it a *hit*. And we use *≥k-HitRate* for showing the performance, which gives the percentage of images out of all test images that achieve ≥ *k hit*.

For subjective evaluation, human evaluators were asked to visually check the images and mark on those tags which they deem as semantically relevant. In addition to *≥k-HitRate*, we also adopted the following statistical metrics from [49] for evaluating the performance: *Mean Reciprocal Rank (MRR)*, which measures where in the ranking the first relevant tag occurs; *Success at rank k (S@k)*, defined as the probability of finding a relevant tag among the top *k* recommended tags; *Precision at rank k (P@k)*, defined as the proportion of retrieved tags that is relevant, averaged over all photos.

3.4.2 Results and Analysis

We ran ten trials with random selections for the training and test sets in order to avoid data selection bias. All experiments were based on these ten trials from the dataset.

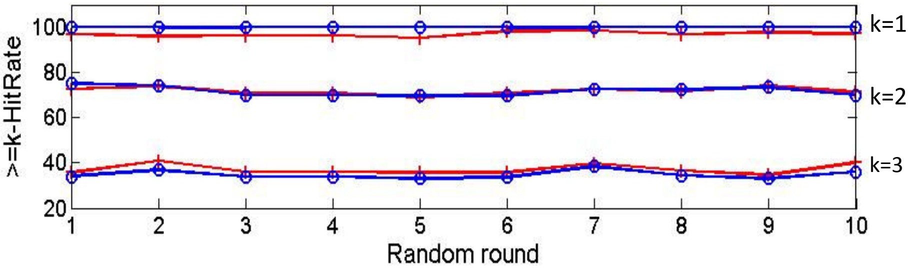


Figure 28: Comparisons of *≥k-HitRate* between different field ranking methods based on objective evaluations: red cross--*a*=0.2; blue circle--*a*=0.5.

Table 4: Tag hit rate of objective evaluation on test sets.

≥*k*-HitRate (%) Average over 10 fixed random rounds	*k*=1	*k*=2	*k*=3
a=0.2	97.0	71.8	37.1
a=0.5	99.9	71.8	34.7

Figure 28 and Table 4 show the results of objective evaluation of the proposed approach when *a* is set as 0.2 and 0.5 respectively (*a* indicates relative contribution of tag popularity score to the overall ranking score. It can be selected based on knowledge of the data source. In our experiments, *a*=0.5 gives relatively best results.). For *k*=1 and 2 cases, both achieved a hit rate close to 100% and above 70% respectively, which are superior to what we achieved in [53] (there is no objective evaluation results reported in [49]). For *k*=1 case, our result is even better than that in [47], with such a much more challenging dataset with high sparsity in tag occurrence. For each selected topic, only a few words appear more than 5 times in the user-provided tags for all the training images. This explains why the rate becomes lower when *k* increases to 2 and 3.

Objective evaluation alone cannot sufficiently evaluate the real performance since many recommended tags are actually good choices for tagging the images although the original users did not use them. If users are offered those recommended tags, they may likely select and use these tags. This is exactly what our tag recommendation system targets at. Therefore, a subjective evaluation is necessary.

In subjective evaluations in this work, three participants were asked to tick all relevant tags in the recommended list for a given image. In order to avoid evaluator bias, each evaluator evaluates only 4 topics (400 test images) from the same random set and only two topics from the same user can be used in one user set. Thus we can have two user sets for this random test set. Except the *≥k-HitRate* metric, we also employ MRR, S@1-5, P@5, P@10 and P@15 metrics as well. Average results for one of the ten random sets under different metrics are listed in Table 5 and Table 6.

Table 5: Tag hit rate of subjective evaluation on one of the test sets.

≥*k* HitRate (%), *a*=0.5	*k*=1	*k*=2	*k*=3	*k*=4	*k*=5
User Set 1	99.2	86.2	64.3	44.0	26.2
User Set 2	99.8	88.7	72.5	48.5	25.0

Table 6: Subjective evaluation on one of the test sets.

S@k (%), *a*=0.5	MRR	S@1	S@2	S@3	S@4	S@5	P@5	P@10	P@15
User Set 1	1.87	71.5	81.7	88.3	91.8	93.2	64.0	35.7	23.8
User Set 2	1.66	78.3	84.8	90.5	93.7	95.3	66.9	35.2	23.5

The subjective evaluation results are statistically better than those of the objective evaluation, which supports our previous argument that, although many generated tags are not listed by the original user, they are good recommendations for the given image. Compared with the state-of-the-art performance in [49], our result is better than the best cases reported. Further, in [49], tag recommendation is purely based text and thus at least one tag from the user must be available; while in our experiment, tags can be recommended based on only images.

We further propose a coarse metric to verify the consistency of subjective evaluation among different users. Consistent rate c between two users based on their selections of relevant tags on the same set of images and the recommended tags is computed as:

$$c = \frac{\sum_{i=1}^{m} \delta_i}{m} \tag{3.13}$$

in which m indicates the number of recommended tags. If the two users agree with each other on the i^{th} tag (i.e., both/neither of them believe the current tag is relevant to the given image), $\delta_i = 1$; otherwise $\delta_i = 0$ (this can be easily extended to multiple users). In our experiments, two different users provided their selections of reverent tags on the m recommended tags (m=15 in our experiments). We picked three topics (i.e., “office”, “Greatwall”, “sunset”) and computed the consistency rates of the two users who provided their selections. The consistent rates of “office”, “Greatwall”, “sunset” topics are 0.94, 0.88 and 0.95 respectively, which indicates high consistency of subjective evaluation among different users. The consistency score of “Greatwall” is relatively lower than those of the other two topics. This is mainly due to the reason that many specific names of locations, such as “simatai”, “mutianyu”, who are not the original author of the pictures could only give their guess about the relevance of a given tag. This reveals a limitation of subjective evaluation with non-author users for photo tag recommendation.

Both objective and subjective evaluation results demonstrate that the proposed approach is capable of capturing the underlying semantic correlation between image contents and text tags.

3.5. Summary

We propose a novel approach for tag recommendation for on-line photos, in which tag recommendation is formulated as a tag ranking problem. All tags from a training set are ranked by a weighted combination of semantic image-tag correlation and tag popularity learnt from the training set. Experimental results based on realistic on-line photos demonstrated the feasibility and effectiveness of the proposed method.

There are many other aspects that can be taken into consideration for further improving the work. For example, other available information of photos, such as title, description, comments, meta-data, etc., can be added as separated features for making tag recommendations, and the image-tag correlation score can be computed by combining the pariwise top correlated instances obtained using these features. In addition, performing semantic grouping on tags before creating the document-term matrix, combining tag co-occurrence strategies proposed in [49], analyzing users' tagging history and social network/activities for providing customized recommendations are also promising directions.

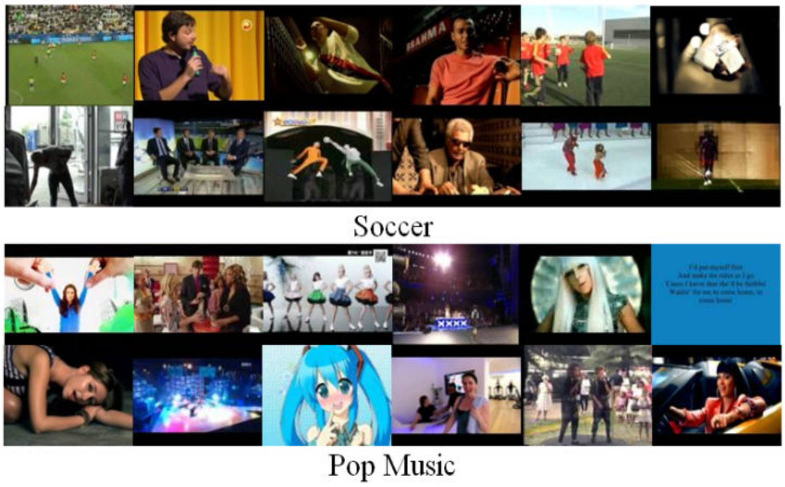
Chapter 4

INCORPERATING PRE-DEFINED TAXONOMY WITH TREE-DRF FORMULATION FOR LARGE-SCALE YOUTUBE VIDEO CLASSFICATION

Automatic categorization of videos in a Web-scale unconstrained collection such as YouTube is a challenging task. A key issue is how to build an effective training set in the presence of missing, sparse or noisy labels. We propose to achieve this by first manually creating a small labeled set and then extending it using additional sources such as related videos, searched videos, and text-based webpages. The data from such disparate sources has different properties and labeling quality, and thus fusing them in a coherent fashion is another practical challenge. We propose a fusion framework in which each data source is first combined with the manually-labeled set independently. Then, using the hierarchical taxonomy of the categories, a Conditional Random Field (CRF) based fusion strategy is designed. Based on the final fused classifier, category labels are predicted for the new videos. Extensive experiments on about 80K videos from 29 most frequent categories in YouTube show the effectiveness of the proposed method for categorizing large-scale wild Web videos.

4.1. Background and Overview of the Proposed Approach

On-line services for archiving and sharing personal videos such as YouTube have become quite popular in recent years. Automatic categorization of videos is important for indexing and search purposes. However, it is a very challenging task for such a large corpus of practically unconstrained (wild Web) videos. A lot of efforts have been devoted to video analysis in the past, but most existing works use very limited number of videos or focus on specific domains such as news, sports etc. Due to practically unbounded diversity of Web videos in both content and quality (as illustrated in Figure 29, analysis of such data is much more challenging than relatively clean videos expected by most existing techniques. A recent study by Zanetti et al. showed that most existing algorithms did not perform well on general Web videos [90]. It also pointed out that one of the major challenges in Web video categorization is the lack of sufficient training data. Manually labeling videos is both time-consuming and labor intensive -- on one hand one has to watch part of a video before (s)he can suggest labels; on the other, web videos are extremely diverse in nature, thus even for human experts, summarizing the video content by using a few keywords is not an easy task.



Soccer

Pop Music

Figure 29: Examples of wild YouTube videos showing extremely diverse visual content.

In this work, we propose a novel approach that combines multiple data sources for wild YouTube video categorization. Starting from a small number of manually labeled samples (as few as 50 per category), we expand the training set by propagating labels to their co-watched videos, collecting data by using internet video search engines (such as Google video search), and even incorporating data from other domains (e.g., text-based webpages). These additional data sources are first pariwise combined with manually-labeled data and a classification model is trained for each combination. For fusing these trained models, we propose a CRF-based tree-DRF fusion approach, which views the taxonomy tree as a random field. Each node (i.e. a category) is associated with a binary label and the output likelihoods of the trained models (applied on the training data) are used as local observations for the nodes. Unlike a traditional fusion strategy that treats each category independently, tree-DRF makes the final labeling decision as a whole by explicitly taking the hierarchical relationships among the categories into consideration. This is crucial to achieve good performance since the data from additional sources is usually quite noisy. The hierarchical relationships among categories provides powerful context for alleviating the noise. Results from extensive experiments on 80K YouTube videos demonstrate that the proposed solution outperforms existing methods that either use just a single data source or traditional data fusion strategy. The main contributions of this work can be summarized as follows: First, to the best of our knowledge, this is the first work that deals with categorization of unconstrained Web videos at such a large scale. Second, we propose a novel approach for integrating data from multiple disparate sources for classification given insufficient training data. Finally, we introduce a tree-DRF based fusion strategy that exploits the hierarchical taxonomy over categories and effectively deals with noise in multiple data sources. It significantly outperforms other commonly used fusion strategies based on SVM and iterative co-training [67, 68, 73].

The rest of this section is organized as follows. We first review the related literature in Section 4.2 followed by the description of multiple data sources we use in Section 4.3. The proposed solution with pariwise data combination and tree-DRF based fusion strategy is presented in Section 4.4. Extensive experimental results, comparisons and analysis are reported in Section 4.5. We conclude in Section 4.6 with a brief discussion on future work.

4.2. Related Work

Compared to image analysis, research on video analysis has been relatively recent. Most existing approaches are either limited to some specific domains (e.g. movies [69, 77], TV videos [70, 86, 89] etc.) or focus on certain predefined content such as human face [70, 84] and human activities [79]. However, large scale categorization of wild Web videos still remains an unsolved problem. The works of Schindler et al. [85], VideoMule [82] and Zanetti et al. [90] are among the initial efforts in this direction. Schindler et al. tried video categorization on 1500 user uploaded videos from 15 categories using bag-of-words representation. However, the classification performance is very poor on this general video set (best classification accuracy is 26.9%). Ramachandran et al. proposed VideoMule, a consensus learning approach to multi-label YouTube videos classification using YouTube categories. Specific amount of data and categories were not reported in their work. Zanetti et al. explored existing video classification methods on about 3000 YouTube videos in their recent work [90]. They pointed out that a major difficulty in Web video analysis is the lack of enough labeled training data. Semi-supervised machine learning approaches [92] are useful for expanding training data in general. However, graph-based methods are used commonly for semi-supervised learning e.g., [93] and semi-supervised SVM [66] are inefficient for large amounts of data with high-dimensional features. Popular co-training/self-training approaches [67, 68, 73] are also typically expensive and their performance is quite sensitive to the amount and quality of the initial training set. Another possible way of collecting more training data is to make use of data from other sources including different domains. It is worth noting that combining multiple data sources is more challenging than combining multiple views of the same data [67, 68, 73], since properties of different data sources are typically more diverse. Multiple data sources can be combined with either early fusion or late fusion strategies [87]. Typically, early fusion assumes that all the features are available for each video, which is not valid in our case (e.g. webpage data has only text features). In late fusion, classifier models are first trained separately; then the trained models are applied to the training set. At the fusion stage, obtained likelihoods from different models are concatenated for each sample and used as a feature vector. Another round of training is then carried out on the new 'features'. Traditional fusion methods are based on regular learning algorithms (such as SVM, AdaBoost), which treat each category independently. On the contrary, given a hierarchical taxonomy over categories, it is desirable to exploit such relationships to achieve robust classification. In this work, we propose tree-DRF to handle the category structure while doing late fusion and empirically show the benefits of such approach.

4.3. Multiple Data Sources

As mentioned earlier, lack of labeled training data is a main bottleneck for general Web video categorization. To alleviate this problem, we first manually labeled 4345 videos from all the 29 categories as initial seeds. This set is further expanded by including samples from related videos, searched videos and cross-domain labeled data (i.e. text webpages), as illustrated in Figure 30. Details of each data source are given below.

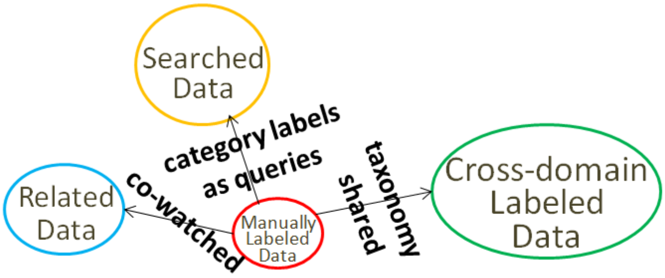


Figure 30: Multiple data sources for YouTube videos including a small set of manually labeled data, related (e.g. co-watched video data), searched data collected by using a video search engine with categories as queries, and cross-domain data (e.g. webpages) which are labeled with the same taxonomy structure.

4.3.1 Manually-labeled Data

To collect the initial seeds for training, we first build a category taxonomy with the help of professional linguists. About 1000 categories are defined using a hierarchical tree of 5 vertical levels (Depth-0 to Depth-4 from top to bottom, Depth-0 is the root). Randomly selected YouTube videos that have been viewed more than a certain number of times are labeled by professionally-trained human experts based on the established taxonomy. Each video is labeled from Depth-0 to the deepest depth it can go. For example, if a video is labeled as *Pop Music*, it must be associated with label *Music & Audio* and *Art & Entertainment* as well. Note that this is a general taxonomy instead of being designed for YouTube videos specifically. Thus, it is not surprising that the distribution of manually-labeled videos over all categories is extremely unbalanced. For example, the *Art & Entertainment* category contains close to 90% of all the labeled videos, and categories such as *Agriculture & Forestry* have only a few videos. In fact, such imbalance reflects the real distribution of videos in the entire YouTube corpus. In this work, we work on 29 categories that had a reasonable amount of manually-labeled samples, i.e., more than 200 for Depth-1 categories and more than 100 for Depth-2 to 4 categories. Manually-labeled samples from these 29 categories (4345 samples in total) cover close to 80% of all the data we labeled, roughly implying that the categories we are working with cover 80% of all possible videos on YouTube. To the best of our knowledge, this is the first work which deals with general Web video classification on such diverse categories. In our experiments, 50% randomly selected samples are used as initial seeds for training (denoted as "M") and the remaining 50% are used for testing.

4.3.2 Related (Co-watched) Data

To increase the training samples for each category, we considered co-watched videos, i.e., the next videos that users watched after watching the current video. We empirically noticed if a video is co-watched more than 100 times with a certain video, they tend to have the same category. Of course, such labels can be noisy but our tree-DRF based late fusion method is able to handle such noise robustly. So, in our experiments, co-watched videos (denoted as "R") of all the initial seed videos with co-watch counts larger than 100 (3277 video in total) are collected to assist training.

4.3.3 Searched Data

Another possibility for expanding the training set is by searching for videos using online video search engines using a category label as a text query. For example, returned videos by submitting query “soccer” may be used as training samples for the “soccer” category. Constrained by the quality of existing search engines, searched videos may be noisy. In our work, we keep about top 1000 videos returned for each category. Since the categories form a hierarchical structure, the videos returned for categories at lower levels are included for their ancestors as well. Querying Google video search gave us a set of about 71,029 videos (denoted as “S”).

4.3.4 Cross-domain Labeled Data

Compared to video labeling, assigning labels to other types of data (e.g. text-based webpages) is usually easier. Although such data comes from a completely different domain, it can be helpful for video classification as long as the samples are labeled using the same taxonomy. This is because we also use text-based features to describe each video as explained in Section 4.1. We collected 73,375 manually-labeled webpages (denoted as “W”) as one of the additional data sources in our experiments.

4.4. Learning from Multiple Data Sources

In Section 4.3, in addition to the manually-labeled data, we introduced several auxiliary sources which may be useful for boosting the video classification accuracy. The main challenge is how to make use of such diverse set of data with different properties (e.g., video content features are not available for web pages) and labeling quality (e.g., labels of searched and co-watched data are fairly noisy). In this work, we propose a general framework to integrating data from mixed sources. As illustrated in Figure 31, each auxiliary data source is first pariwise combined with the manually-labeled training set. Initial classifiers are trained on each such pair. For each pair, two separate classifiers are learned, one with text-based and another with content-based features. For example, in Figure 31 M_{Sc} is a content-based and M_{St} is a text-based model for the combination of manually-labeled data and searched data. Trained models are then fused using a tree-DRF fusion strategy. Different from traditional methods that fuse models for each category independently, the proposed tree-DRF incorporates the hierarchical taxonomy structure exploring the category relationships effectively. Next we introduce the features used for training individual classifiers followed by the description of our tree-DRF fusion method.

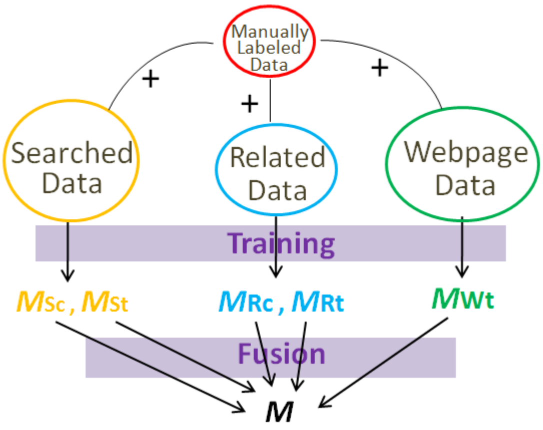


Figure 31: General framework of the proposed solution: Additional data sources are first combined with manually-labeled data independently and classifier models are trained based on either text or content features for each combination. Individual classifiers are further fused to form the final classifier *M*.

4.4.1 Features

It is well known that designing good features is perhaps the most critical part of any successful classification approach. To capture the attributes of wild Web videos as completely as possible, state-of-the-art text and video content features are utilized in our experiments as briefly summarized below.

Text features: For each video, the text words from title, description and keywords are extracted. Then, all these words are weighted to generate text clusters. The text clusters are obtained from Noisy-Or Bayesian Networks [81], where all the words are leaf nodes in the network and all the clusters are internal nodes. An edge from an internal node to a leaf node means the word in the leaf node belongs to that cluster. The weight of the edge means how strongly the word belongs to that cluster.

Video content features: color histogram computed using hue and saturation in HSV color space, *color motion* defined as cosine distance of color histograms between two consecutive frames, *skin color* features as defined in [74], *edge features* using edges detected by Canny edge detector in regions of interest, *line features* using lines detected by probabilistic Hough Transform, *histogram of local features* using Laplacian-of-Gaussian (LoG) and SIFT [80], *histogram of textons* [78], *entropy features* for each frame using normalized intensity histogram and entropy differences for multiple frames, *face features* such as number of faces, size and aspect ratio of largest face region (faces are detected by an extension of AdaBoost classifier [88]), *shot boundary* detection based features using difference of color histograms from consecutive frames [91], *audio features* such as audio volume and 32-bin spectrogram in a fixed time frame centered at the corresponding video frame, *adult content features* based on a boosting-based classifier in addition to frame-based adult-content features [83]. We extract the audio and visual features in the same time interval. Then, a 1D Haar wavelet decomposition is applied to them at 8 scales. Instead of using the wavelet coefficients directly, we take the maximum, minimum, mean and variance of them as the features in each scale. This multi-scale feature extraction is applied to all our audio and video content features except the histogram of local features [72]. Note that features are not the main contribution of this work. Due to space limitation, we skip the details of the features and refer the reader to the respective references. For fair comparisons, all the experimental results reported in this work are obtained based on the same set of features.

4.4.2 CRF-based Fusion Strategy

Conditional Random Fields (CRFs) are graph-based models that are popularly used for labeling structured data such as text [76] and were introduced in computer vision by [75].

In this work, we use outputs of discriminative classifiers to model the potentials in CRFs as suggested in Discriminative Random Field (DRF) formulation in [75]. We denote the observations as $\boldsymbol{y}$ and the corresponding labels as $\boldsymbol{x}$. According to CRFs, the conditional distribution over labels given the observations is defined as a Gibbs field:

$$p(\boldsymbol{x} \mid \boldsymbol{y}) = \frac{1}{Z} \exp(\sum_{i \in S} A_i(x_i, \boldsymbol{y}) + \sum_{i \in S} \sum_{j \in N_i} I_{ij}(x_i, x_j, \boldsymbol{y}) \tag{4.1}$$

where S is the set of all the graph nodes, N_i is the set of neighbors of node i, and Z is a normalizing constant called partition function. Terms A_i and I_{ij} are the unary and pariwise potentials sometimes referred to as *association potential* and *interaction potential* respectively.

4.4.3 Tree-DRF

As discussed earlier, in this work we use multiple data sources that are combined by a late fusion step. We want a fusion strategy that can combine the classifier outputs from different sources while respecting the taxonomy over categories. The DRF framework described above gives a natural way of achieving that. Formally, A_i learns to fuse the outputs of independent classifiers while I_{ij} enforces the category relationships defined by the hierarchical taxonomy. In [75], DRF is used for image classification, in which a graph is built on image entities, i.e., pixels or blocks. On the contrary, in our case, the graph is defined over the hierarchical taxonomy (i.e., a tree over categories) and a node represents a category. Each node i is associated with a binary label variable x_i, i.e., $x_i \in \{-1,1\}$ implying whether i^{th} category label should be assigned to the input video or not. The scores from different classifiers for the i^{th} category on a given video are concatenated in a feature vector, which serve as the observation y_i. Figure 32 illustrates the proposed tree-DRF.

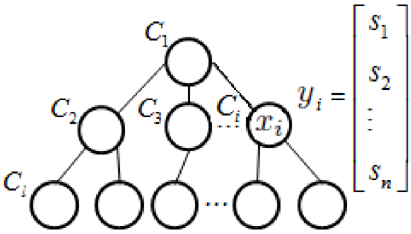

C_1
C_2
C_3
C_i
x_i
C_i
...
$y_i = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{bmatrix}$

Figure 32: Late fusion strategy based on tree-DRF. For each input video, a tree-structure over categories is defined. The binary label at the i^{th} node (x_i) represents whether that video should be assigned the category label C_i The observation vector (y_i) is simply the concatenation of classifier scores on the video for that category.

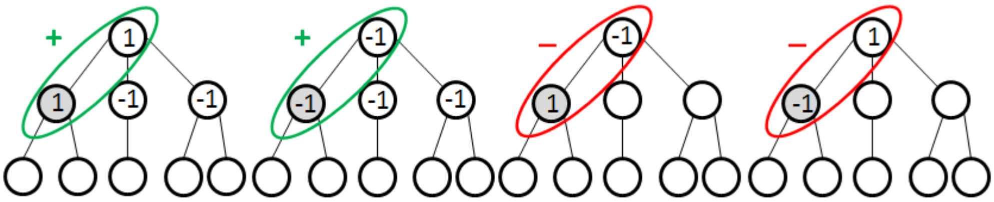


Figure 33: Tree-structure interaction (current node x_i is shaded): consistent labels of parent-child pairs make positive contribution; while likelihoods are penalized for inconsistent cases.

Following [75], *association potential* is defined as,

$$A_i(x_{i,}\boldsymbol{y}) = \log(\frac{1}{1+\exp(-x_i \boldsymbol{w}_i{}^T h_i(\boldsymbol{y}))}) \qquad (4.2)$$

where $\mathbf{w}_i$ is a parameter vector and $h_i(\mathbf{y})$ is a feature vector at site i. We define $h_i(\mathbf{y})$ to include the classifier scores and their quadratic combinations. Note that unlike the homogeneous form used in [75], the association potential in our tree-DRF model is inhomogeneous. There is a separate association parameter $\boldsymbol{w}$ for each node. The reason is that since a different set of classifiers is learned for each category (i.e., a node), forcing the weight vectors defining combinations of such disparate sets of classifiers to be the same for all the nodes is too harsh. Thus, we allow the model to choose a different weight vector for each category. Of course, it leads to more parameters in the model but since our graph is fairly small (just 29 nodes), and the size of observation vector, i.e., the number of classifiers, is also small, the computational overhead was negligible. Moreover, overfitting is also not a concern since we have enough training data for such small number of parameters.

The *interaction potential* in tree-DRF is defined as,

$$I_{ij}(x_i, x_j, \boldsymbol{y}) = x_i x_j \boldsymbol{v}^T \mu_{ij}(\boldsymbol{y}),\ j \in N_i \tag{4.3}$$

where **v** are the model parameters and $\mu_{ij}(\boldsymbol{y})$ is a pariwise feature vector for nodes *i* and *j*. In this work, we only explored data-independent smoothing by forcing $\mu_{ij}(\boldsymbol{y})$ to be a constant. Similarly, the parameter ***v*** was kept to be the same for all the node pairs. One can easily relax this to allow directional (anisotropic) interactions between parents and children which can provide more powerful directional smoothing. We plan to explore this in the future.

We used the standard maximum likelihood method for parameter learning in tree-DRF. Since the graph structure is a tree, exact unary and pariwise marginals were computed using Belief Propagation (BP). For inference, we used sitewise Maximum Posterior Marginal (MPM), again using BP. Results of tree-DRF fusion and comparisons to regular fusion strategy based on SVM and Co-training are presented in Section 4.5.

4.5. Experiments and Results

In order to verify the effectiveness of the proposed solution, we performed extensive experiments with about 80K YouTube videos and about 70K webpages. We first introduce the experimental data and settings in the next section followed by a brief description of the evaluation metric.

4.5.1 Experimental Data and Setting

As described in Section 4.3, four different data sources and 29 major categories are used in our experiments. The categories followed by their path in the taxonomy tree are: “Arts

& Entertainment” (1), “News” (2), “People & Society” (3), “Sports” (4), “Celebrities & Entertainment News” (1, 5), “Comics & Animation” (1, 6), “Events and Listings” (1, 7), “Humor” (1, 8), “Movies” (1, 9), “Music & Audio” (1, 10), “Offbeat” (1, 11), “Performing Arts” (1, 12), “TV & Video” (1, 13), “Team Sports” (4, 14), “Anime & Manga” (1, 6, 15), “Cartoons” (1, 6, 16), “Concerts & Music Festivals” (1, 7, 17), “Dance & Electronic Music" (1, 10, 18), “Music Reference” (1, 10, 19), “Pop Music” (1, 10, 20), “Rock Music” (1, 10, 21), “Urban & Hip-Hop” (1, 10, 22), “World Music” (1, 20, 23), “TV Programs” (1, 13, 24), “Soccer” (4, 14, 25), “Song Lyrics & Tabs” (1, 10, 19, 26), “Rap & Hip-Hop” (1, 10, 22, 27), “Soul & R&B” (1, 10, 22, 28), and “TV Reality Shows” (1, 13, 24, 29). In our experiments, binary classifiers are trained for each category respectively. Content features and text features are trained separately by using AdaBoost and SVM, respectively. LibLinear [71] is used to train SVMs when training samples exceed 10K. Trained models are then integrated using regular SVM based late fusion strategy [87]. Since webpage data has only text features (no content features), only a single model is learned for this set. The training data from two sources (i.e., manually-labeled data plus one additional data source) is combined before training the classifiers. After all the data sources are leveraged, fusion is performed for content and text features for three pariwise combinations, represented by five individual classifiers. In the training process, negative training samples for each category are randomly selected from other categories with a negative-positive ratio of 3:1.

4.5.2 Evaluation Metrics

While testing, since binary classifiers are trained for each category, each test sample receives 29 classification decisions (either “yes” or “no”). Multiple labels for a single sample are allowed. As the category labels form a taxonomy structure, predicted categories/labels are also propagated to their ancestors as done while generating ground-truth labels for the training data. For example, if a test sample has a ground-truth label

"Art & Entertainment" / "TV & Video" / "TV Programs", it is treated as a true positive sample for "Art & Entertainment" category if it is classified by any of these three classifiers. For the quantitative evaluation, we compute Precision, Recall and F-score. To perform aggregate assessment of the classification performance, we also compute F-scores for each depth level of the taxonomy.

4.5.3 Results and Analysis

The objective of the proposed approach is to improve video classification performance by making use of data from multiple sources of varied quality. Table 7 lists classification accuracy of each data source (due to space limitation, we only show F-score in all tables and figures). Performance with just the related videos (R) or the searched videos (S) is much worse than that from manually-labeled data (M). It shows that neither related videos nor searched videos are sufficient for training a reliable classifier. Webpage data (W) obtained from a completely different domain, which does not even contain video content, works better than manually-labeled data for most taxonomy depths. This is possible since even noisy text based features for videos are usually more reliable than video content features.

In order to achieve better results, we combine each of the additional data sources pariwise with manually-labeled training data. As shown in Table 8, for related video source, pariwise combination achieves significant improvements over just using related videos and even better than training on manually-labeled data. For the searched videos, performance of pariwise combination is also better than that for just the searched data, but worse than that of the manually-labeled data. In terms of the webpage data, pariwise combination is not always superior to the single sources. Overall, there are two observations: 1) Pariwise combination with manually-labeled data can improve classification accuracy of any single additional source in most cases; 2) Introducing additional data sources by simply merging them with the manually-labeled data does not guarantee improvement for all cases over the baseline configuration, i.e., using just the manually-labeled data for training.

Table 7: Classification accuracies of single data sources.

F-scores	Depth 1	Depth 2	Depth 3	Depth 4
M	0.80	**0.60**	0.45	0.41
R	0.74	0.53	0.37	0.34
S	0.73	0.51	0.37	0.31
W	**0.84**	0.54	**0.48**	**0.45**

Table 8: Classification accuracies of each single data sources combined with manually-labeled data.

F-scores	Depth 1	Depth 2	Depth 3	Depth 4
M + R	**0.86**	**0.63**	**0.47**	**0.49**
M + S	0.78	0.57	0.43	0.37
M + W	0.84	0.55	0.45	0.39

Table 9: Classification performance of fusing pariwise combinations of data using different fusion strategies.

F-scores	Depth 1	Depth 2	Depth 3	Depth 4
All, SVM	0.84	0.65	0.46	0.49
All, Tree-DRF	**0.87**	**0.72**	**0.57**	**0.52**
M+R, Tree-DRF	0.85	0.66	0.48	0.45

Next, we fuse the single classifier models trained from pariwise combinations to further boost the classification performance. First row of Table 9 shows the results of using regular SVM late fusion strategy. Compared to the best cases in Table 8, fusing all data sources does not achieve any obvious improvement (for Depth-1 and Depth-3, results are even worse). It is because, for SVM, when the feature dimension increases but not the amount of training data, the test performance may degenerate due to over-fitting. This observation underscores our previous assertion that an inappropriate fusion strategy for adding unreliable data sources may even harm the classification accuracy.

Results of the proposed tree-DRF fusion strategy are reported in Table 9-second row. For all taxonomy depths, tree-DRF outperforms regular SVM fusion. Especially for Depth-2 and Depth-3, in which the categories can benefit from both parent categories and child categories, it achieves 0.07 (11%) and 0.11 (24%) improvements in F-scores. Compared to the baseline performance (Table 7-first row), it gains 0.07 (9%), 0.12 (20%), 0.12 (27%), 0.11 (27%) F-score improvements for Depth-1 to Depth-4 respectively. Such significant improvements are due to the taxonomy tree based learning of tree-DRF. In other words, since interactions between parent and child nodes are considered, noise in the additional data sources can be largely filtered. This is because useful information is typically consistent for neighboring nodes and thus can be emphasized by the *interaction potential* in tree-DRF.

For analyzing the effectiveness of including additional data sources, we applied tree-DRF on the pair of manually-labeled data and related data (which gave the best results among all pariwise combinations with regular fusion of content models and text models) in the third row of Table 9. Compared to tree-DRF on all data (second row in Table 9), results are worse, which demonstrates the gain from multiple data sources by using tree-DRF. For easy comparison, accuracies from all experiments are summarized in Figure 34.

To analyze the results for individual categories, we illustrate F-scores for the baseline method (i.e., using only manually-labeled data for training), and SVM and tree-DRF based fusion with all data sources in Figure 35. For most of the categories, tree-DRF outperforms the other two methods, especially for the categories with small amount of training samples but relatively large number of neighbors.

In addition to SVM and tree-DRF based fusion, we also conducted experiments with co-training on different combinations of the four data sources with different settings (e.g. by varying the number and weights of new training samples added in each iteration, and the stopping criteria). In the best case, F-scores for Depth-1 to Depth-4 were 0.82, 0.61, 0.44 and 0.40 respectively, which are much lower than the proposed tree-DRF method and even lower than regular SVM fusion strategy. Regarding computational complexity of tree-DRF, since the graph is built on the taxonomy, it results in a very small graph having just 29 nodes connected with very sparse edges. Also, since the outputs of individual classifiers are used as features, it leads to very low-dimensional features. Hence, overall the tree-DRF is extremely fast in training as well as testing.

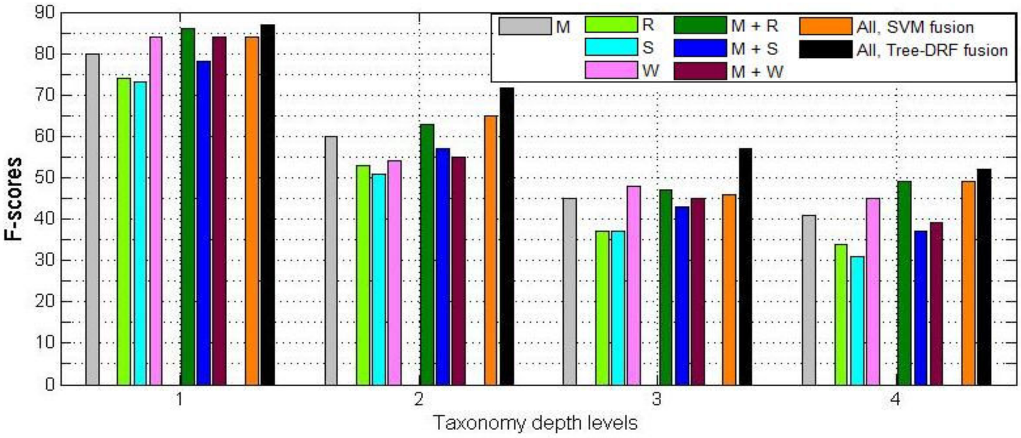


Figure 34: Comparison of classification accuracies from different data sources and combinations. Tree-DRF with all pariwise data combinations achieved the best performance. M: Manually-labeled data, R: Related Videos, S: Searched Videos, W: Webpage data.

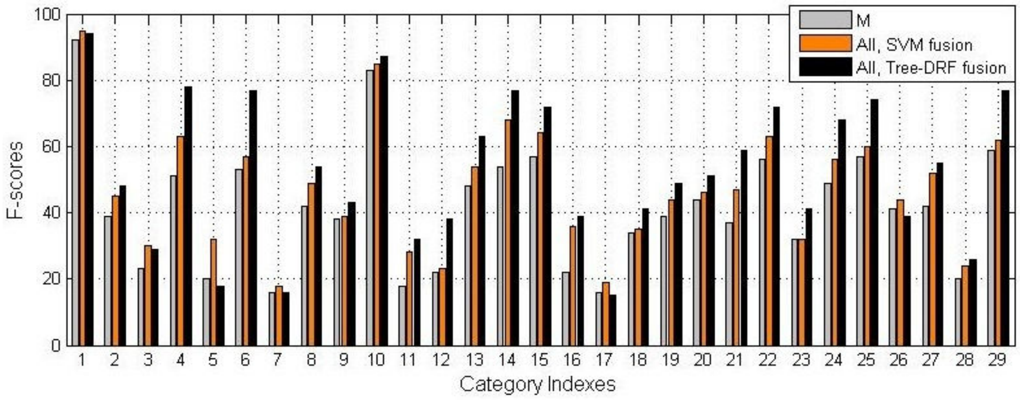


Figure 35: F-scores of 29 categories on manually-labeled data (M), all data with SVM fusion and all data with tree-DRF fusion. Tree-DRF performed better than the other two methods for most categories.

4.6. Summary

In this work, we proposed a novel solution to wild web video categorization on a large-scale dataset (more than 80 thousand YouTube videos). Our approach provides an effective way of integrating data from diverse sources, which largely alleviates a major problem of lack of labeled training data for general web video classification. Tree-DRF was proposed for fusing models trained from individual data sources when combined with small amount of manually-labeled data in a pariwise fashion. Compared to traditional fusion strategies, the proposed tree-DRF takes the taxonomy tree of category labels into account, resulting in significant improvement in classification performance. Experimental results on a large-scale YouTube dataset show that the proposed approach is effective for categorizing wild videos on the Web.

Currently we only consider undirected relationships between parent and child categories in tree-DRF. More sophisticated anisotropic formulations of interaction potential for parent or child neighbors, and siblings may further improve the labeling performance. In addition, it is also possible to make use of unsupervised learning methods (e.g. clustering) for assigning weights to noisy labeled samples and adjusting their contributions accordingly while training classifiers. Integrating an iterative co-training framework of incrementally adding additional unlabeled data is also a possible way of further expanding the training data set and improving the classification performance.

Chapter 5

LEVERAGING VIDEO TEMPORAL ORDER IN SPARSE REPRESENTATION FOR CONSUMER VIDEO SUMMARIZATION

Automatic video summarization is critical for facilitating fast browsing and efficient management of multimedia data. Compared to well-edited videos with predefined structures (e.g., movies) or constrained contents (e.g., news or sports videos), upon which existing methods focus, the main challenges of summarizing unconstrained amateur or consumer videos include dealing with extremely diverse contents without any pre-imposed structure and poor video quality due to camera shake. To address these challenges, we explore a signal-reconstruction based approach relying only on visual content. In particular, we propose a sequence-kernel-based sparse representation approach for directly summarizing consumer videos. A dictionary of subsequences is first constructed from clustered frames with importance ranking scores of extracted high-level semantics. Video summarization is formulated to seek an optimal combination of the dictionary elements that robustly represents the original video. Weighted-sequence distance is exploited to compute the approximation error, and the kernel-based feature-sign algorithm is used to estimate the sparse coefficients. A linear combination over the dictionary with the obtained optimal sparse coefficients is output as the final summary video. Extensive experiments are performed on 71 videos with ratings from 7 evaluators. Results obtained by the proposed approach compare favorably with two existing methods both visually and quantitatively, validating its effectiveness.

5.1. Background and Overview of the Proposed Approach

With a rapid growth in the use of digital cameras and camcorders, personal or consumer videos from amateur users have become one of the major sources of multimedia contents. Automatic video summarization techniques are urgently needed for fast and efficient browsing, managing and sharing the huge amount of video data. The video summarization problem has been investigated for years, while it mainly focused on structured and clean videos (e.g., news and sports videos). Compared to professionally shot and well-edited videos, unconstrained consumer videos record extremely diverse contents and are often referred to as videos "in the wild". Also, such videos often lack a pre-imposed structure (see Figure 36) and may exhibit low quality due to factors such as camera shake and poor lighting (see Figure 37).

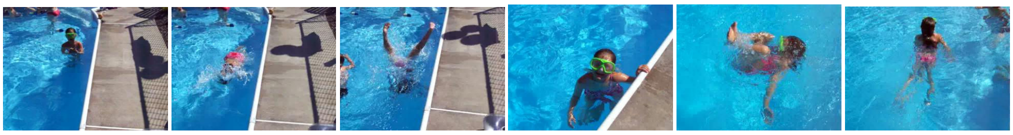


Figure 36: Example – An unstructured home video.

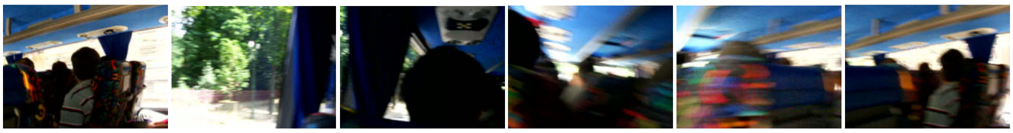
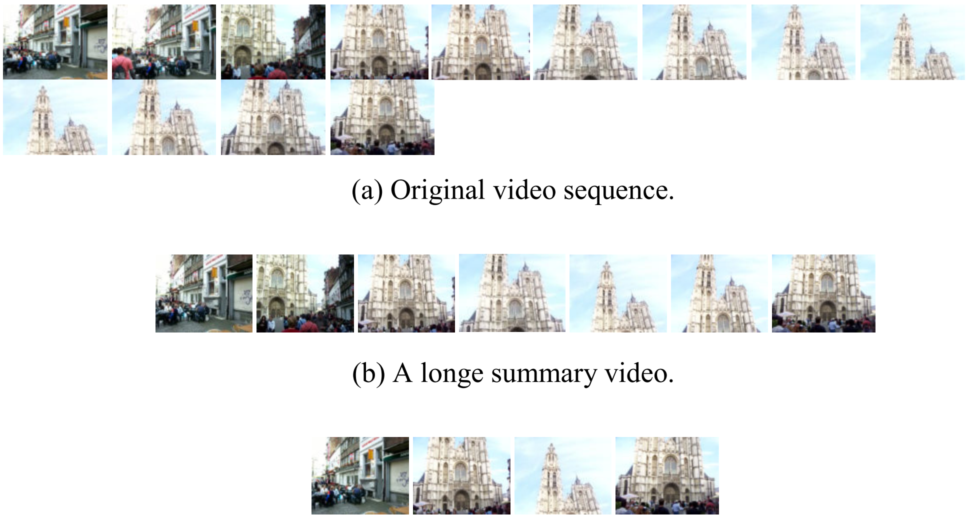


Figure 37: Example – A blurred home video.

This work addresses the problem of automatically summarizing consumer videos based on visual contents (i.e., without sound track or other metadata). Given an unedited video clip with unconstrained content, the approach is expected to generate a summary video (with temporal compression only) of the user-specified length that covers the visual contents as completely as possible and maintain the temporal structure of the original video as well. Figure 38 illustrates an example, where (a) is the original video, and (b) and (c) are two summary videos of different lengths. Both videos retain the visual contents (to different extents) and preserve the temporal structure of the original video - panning from the street, to the crowd, to the bottom of the church, then going up to top of the church and finally back to the bottom. It is worth pointing out that bottom of the church appears twice in the summary video by intent, which reflects the importance of the temporal order of the scene changes (otherwise this scene would be redundant).

Existing approaches, such as [105, 106, 111-114], on video summarization typically follow a procedure of segmenting the original video into clusters of frames, rating the importance of the contents of each cluster and selecting representative key frames/short skims from each cluster to form the final video summary. Without any prior knowledge of the visual contents, such approaches share a significant challenge as to how to determine an appropriate number of segments.

In this work, we propose an approach to directly generate video summaries for unconstrained videos. Given an input video, we first create a dictionary of subsequences (we define a subsequence as a subset of frames or short snippets from the original video with the original temporal order imposed) from the original video. Based on the sparse representation theory [94], the video summarization problem is formulated as seeking a set of sparse coefficients over the dictionary elements that constructs an optimal combination (i.e., a summary video) best representing the original video. Different from most existing applications of sparse representation in which each dictionary element is a vectorized data point, dictionary elements in this work are subsequences with temporal order imposed, which requires a new metric (other than the standard L2 norm) for measuring the reconstruction errors. We adopt edit distance based Weighted-sequence Distance (WSD) [101] with revised operation costs to measure the reconstruction errors, which reflects the quality of candidate summaries. A kernel representation of this measure is further used with the feature-sign algorithm [103, 104] to solve the optimal (approximated) sparse coefficients, which are used for generating the final summary video. To create a dictionary of practical size while well covering the subsequence space (i.e., all possible combinations of a given number of snippets from the original video), we perform over-clustering of the frames from the original video and select short snippets with high importance scores in terms of image quality, face detection, scene complexity, and motion change (details in later sections).



(a) Original video sequence.

(b) A longe summary video.

(c) A short summary video.

Figure 38: Examples of desirable video summarization: (a) the original video; (b) and (c) two summary videos of different lengths.

Advantages of the proposed approach can be summarized from the following four aspects: 1) This approach is designed to generate video summaries directly without key frame extraction and expansion, which avoids the problems of setting the key frame number and expanding key frames back to a summary video; 2) The proposed approach is adaptive to user-specified lengths of summaries via automatically adjusting the sparsity parameter; 3) Instead of using any single frame independently, dictionary elements are defined as subsequences and quality of candidate summaries are evaluated by using sequence-based distance measure, which well retains the temporal structure of the original video; 4) The proposed approach provides a general framework so that high-level semantic information can be incorporated naturally into the dictionary creation step of sparse representation, which is mostly built for signal reconstruction that is low-level in nature.

To evaluate the performance of the proposed approach, we apply it on 71 real consumer videos depicting diverse contents. Rating scores from 7 evaluators on the results generated using the proposed approach and comparisons with three other methods demonstrate the effectiveness and advantages of the proposed approach.

In the remainder of this section, we first briefly review related work on video summarization in Section 5.2. Details of the proposed approach are elaborated in Section 5.3. Experiments, user evaluations, and comparisons are presented in Section 5.4. We summarize the proposed approach, its limitations, and future work in Section 5.5.

5.2. Related Work

Generally, there are two ways to summarize video data: static key frames and dynamic video skims. The former is a set of key frames from the original video [105, 106] that can be printed or displayed as a slideshow; the latter is a condensed version of the original video [111-114] that consists of a series of short clips that concatenate to retain the dynamics and characteristics of the original sequence by some measure. Most existing work follows a procedure of segmenting the original video into clusters, then ranking the contents of each cluster, and further selecting representative frames or sub-skims from each cluster to form the final summary [105, 106, 111-114]. However, this type of approaches suffers from a big problem of having to set an appropriate number of segments, which is unknown in practice. Visual features have been used extensively for video summarization [117, 118], whereas methods exploiting audio [107], and video metadata such as camera motions (e.g., panning, zooming) [105] have also been reported. For videos generated in a controlled fashion (e.g., broadcasting sports and news), the underlying structure can be a strong cue, a case in point is most TRECVID systems [119], which rely significantly on the use of captions, audio, and the underlying structure of broadcast news/sports videos. However, these techniques are challenged by consumer videos in which pre-defined structures is lacking, camera motion information is hard to compute, and audio is often missing or noisy. Therefore, we choose to explore a signal-reconstruction based approach that relies only on visual content.

To reduce redundancies, most prior works perform only temporal compression. Some recent work explore the possibility of summarizing videos both spatially and temporally [109, 115], which leads to a summary skim composed of synthesized images. However, these approaches require sophisticated analysis of the visual content at semantic level, e.g., object detection, image segmentation, and visual saliency detection, which remain open computer vision problems themselves. Furthermore, whether the synthesized frames are acceptable is another unsolved issue. In this work, we only consider removing temporal redundancies from the original video.

For evaluations, subjective user evaluation is commonly followed by most existing works. Due to the time-consuming and labor-intensive nature of the task, most papers report evaluation results for only a few videos (often under 10) from several judges. Kang et al. perform experiments on 30 videos that are extremely short (i.e., 100 frames in total on average) [110], without reporting any quantitative results in the work. In this work, we apply the proposed approach on 71 real consumer videos and collect rating scores from 7 evaluators. Statistics and significance of the ratings are presented and analyzed in the experiment section.

5.3. Summarization with Sequence-Kernel-Based Sparse Representation

Consider video summarization as a problem to select a subset of frames/short snippets (which forms a subsequence) from the original video sequence. If all possible subsequences can be enumerated and there exists a metric to compare each candidate with the original video quantitatively, optimal summary can be obtained by selecting the one with the highest evaluation score. For example, for the video sequence of Figure 38, a few sample candidate subsequences are shown in Figure 39. Each of the candidates is rated in terms of a criterion of how good it covers the visual content and the temporal structure of the original video. Intuitively, the third row is the best one among the four shown because it preserves the major scenes and the temporal order of the scene changes.

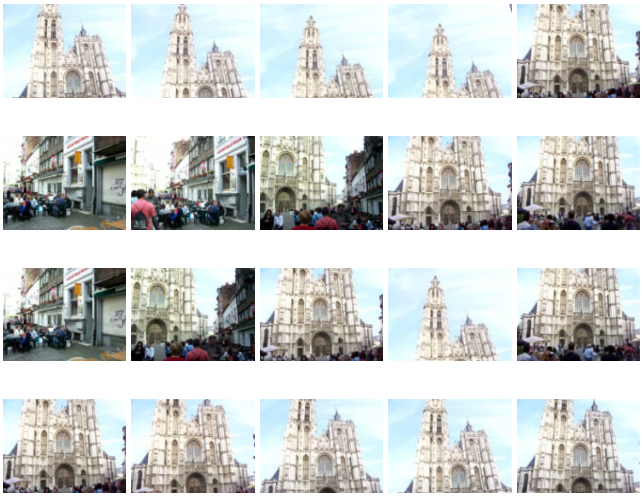


Figure 39: Candidate summaries of the video of Figure 38. The third row is the best subsequence among all the candidates to summarize the original video while preserving temporal order.

5.3.1 Overview of the Proposed Approach

Recently, much interest has been focused on computing linear sparse representation [94] with respect to an over-complete dictionary of a set of basis elements. Suppose we have an underdetermined system of linear equations:

$$y = A \cdot \alpha \tag{5.1}$$

where $y \in R^m$ is the target signal to be approximated, $\alpha \in R^n$ is the vector for unknown reconstruction coefficients, and $A \in R^{m \times n}$ (m<n) is the over-complete dictionary with n bases. Generally, a sparse solution is more robust and efficient for coding and reconstructing the target signal and has been widely used for various vision related applications, such as image restoration [95]. The sparsest solution can be obtained by solving a L_1 optimization problem in polynomial time by standard linear programming method [96]:

$$\min_{\alpha} \| \alpha \|_1, \; s.t. \; y = A \cdot \alpha \tag{5.2}$$

In this work, the input video sequence y that is to be summarized (i.e., the target signal) is represented as

$$y = [f_1, f_2, ..., f_n]^T . \tag{5.3}$$

in which f_i ($1 \leq i \leq n$) is a frame or a short snippet (defined as a group, e.g., 5-10, of consecutive frames from the original video). For the sake of simplicity, here we call it a *snippet.*

In the proposed video summarization approach, we first generate a dictionary A of M elements

$$A = [a_1, a_2, ..., a_M] \tag{5.4}$$

in which each dictionary element a_j is a subset of snippets selected from y (we shall discuss the stragay of selection in Section 5.3.2)

$$a_j = [x_1, x_2, ..., x_n]^T, \ 1 \leq j \leq M \tag{5.5}$$

and a_i contains exactly l non-zero entries (i.e., l snippets from the original video),

$$x_i = \begin{cases} f_i, \ i \in S_j \\ 0, \ \text{otherwise} \end{cases}, \ 1 \leq i \leq n, \ |S_j| = l \tag{5.6}$$

where S_j denotes the set of l indices derived from the input video y to construct i$^{\text{th}}$ dictionary element a_j. Construction of dicitionary elements is discussed in Section 5.3.2. Let y_k be the desired (unknown) summary of the video y . We represent y_k as a sparse linear combination of dictionary elements as shown below:

$$y_k = A \cdot \alpha, \ ||\alpha||_0 = m, \ m << M \tag{5.7}$$

where α is an m-sparse coefficient vector, i.e., only m non-zeros entries are allowed in α. y_k is a linear combination of all dictionary elements in A with non-zeros coefficients while retaining their temporal order.

Since the summary video should represent the salient contents of the input video, therfore, the sparse coefficient vector, α , is estimated by minimizing the error between the the input video y and the summary video y_k as given below:

$$\alpha_0 = \arg\min ErrFn\left(y, y_k\right), \ \|\alpha\|_1 < m \tag{5.8}$$

where $ErrFn(\cdot,\cdot)$ compares the two inputs and estimates the error. Typically, L_2-norm is used in such cases. But, in this case, $ErrFn(\cdot,\cdot)$ needs to be selected carefully as y and y_k are sequences rather than regular vectorized data points, standard L_2 norm is no longer applicable for computing the reconstruction error here.

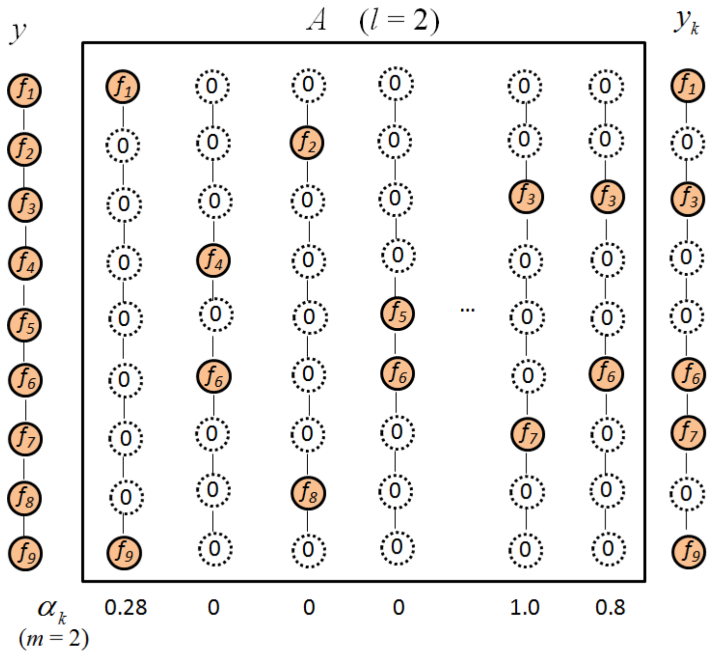


Figure 40: An illustration of generating a candidate video summary from a dictionary of subsequences.

Figure 40 illustrates an example, where y on the left is the original video which contains 9 snippets. A , in the middle, is a dictionary of subsequences (each dictionary element has 2 snippets from y). With 2-sparse (i.e., m=2) coefficients α_k , a candidate summary y_k is constructed on the right.

There are three major problems involved in the above formulation: (1) How to create a good dictionary with a practical size of subsequences while well covering the *subsequence space*. (2) How to quantitatively measure the reconstruction error in Eq.

(5.8) (i.e., quality of each candidate subsequence). (3) How to efficiently search for the optimal combination (i.e., sparse coefficients) of the dictionary elements which can be used for generating the final summary. We shall elaborate these three problems in the following subsections.

5.3.2 Dictionary creation

Given a video, enumerating all possible subsequences of frames or short snippets and evaluating their quality in terms of summarizing the original video is a problem of combinatorial complexity. In addition, based on the definition of the distance measure, there is no systematic way to directly construct an optimal summary without fully searching the candidate space.

In this work, we propose to adopt a dictionary based approach, in which the final summary video is constructed by combining sparsely selected dictionary elements. However, to retain the temporal order of the snippets, dictionary elements must contain at least a few snippets. For a long original video, the *subsequence space* is still far from affordable. Thus, it is necessary to have an appropriate sampling scheme to reduce the search space that guarantees, to some extent, the quality of the representative candidates.

Since a summary video is a fast skim of the original video, its temporal redundancy needs to be minimized. To this end, we proposed to generate dictionary elements through temporal over-clustering of frames/snippets. Representative frames/snippets from each cluster are used to construct a dictionary element. Intuitively, this strategy groups similar frames/snippets, which makes the constructed dictionary element cover as many different scenes as possible, leading to high-quality candidates.

In this work, we first perform temporal over-segmentation of the original sequences by spectral clustering [98] of the frames/snippets while preserving the temporal order. To generate a dictionary element, clusters are first selected as sources. Within each cluster, frames/snippets with top importance scores are selected to construct the dictionary element. Note that the number of clusters is a tradeoff between shrinking the search space and risking a loss of some scenes. In our experiments, it is specified based on the length of the original video (refer to the experiment section for details). We consider four aspects to define the overall importance score of frame/snippet f_i.

Face detection score is computed as an accumulated production of the size of the face (x_j denotes the diagonal length of the j^{th} detected face region) and the relative position of the face center (d_j indicates the distance of the center of the detected face region to the center of the frame (we used Viola and Jones's human face detector [100] in our experiments)

$$a_i^{(F)} = \sum_{j=1}^{J} x_j \cdot d_j \tag{5.9}$$

Image quality score computes *BIQI* image quality score [99] (the larger the better image quality)

$$a_i^{(Q)} = \exp(-BIQI(f_i)) \tag{5.10}$$

Scene complexity score of a single frame is defined as its Kolmogorov complexity measured by the normalized (to the frame size) length of its Bzip2 sequence [107].

$$a_i^{(C)} = BZIP2(f_i) / size(f_i) \tag{5.11}$$

Motion change score is defined as the Euclidean distance between the histograms of the motion magnitudes (i.e., Lucas-Kanade optical flow motion magnitudes estimated over a dense grid) of the current frame and the $(i+\tau)^{th}$ frame

$$a_i^{(M)} = \| H_i - H_{i+\tau} \|^2 \tag{5.12}$$

where τ is a temporal offset, set as 10 in our experiments.

The above scores are normalized to [0, 1] for the giving video before further computation. Final frame important score is defined as a weighted summation of the above four scores

$$a_i = \eta \cdot [a_i^{(Q)}, a_i^{(F)}, a_i^{(C)}, a_i^{(M)}]^T \tag{5.13}$$

where η is a vector of constant weights for balancing the four components (score of a representative frame is used for a snippet).

In generating a dictionary element, clusters are first selected as sources. Within each cluster, frames/snippets with top importance scores are selected to construct the dictionary element. Number of clusters is a tradeoff between shrinking the search space and risking a loss of some scenes. In our experiments, it is specified based on the length of the original video (refer to the experiment section for details).

Furthermore, these four scores are normalized to [0, 1] and the important score for a frame is computed by taking a linear combination of all the four normalized scores for that frame. In the current implementation, we assign equal weight to each score for simplicity. Due to space limitation, we omit detailed discussions of the four scores.

5.3.3 Weighted-sequence Distance

As we mentioned previously, in our formulation of sparse representation, the two signals for comparison (i.e., the original video y and a candidate summary y_k) are sequences with dramatically different lengths (due to the nature of video summarization). Classic distance metrics, such as Euclidean distance, Cosine distance, etc, are obviously not applicable. In this work, we propose a generalized version of the classical Levenshtein Distance (also known as String Edit Distance [120]), which takes both the attribute and the character (codeword) distances into consideration.

String distance/similarity problems widely appear in many areas of computer science. For example, in Web search, string matching/text comparison is one of the basic problems for text-based retrieval (e.g., [121, 122]); in computational biology, string matching techniques are often used for comparing biological patterns (e.g., [123]). Edit distance, a basic string similarity metric, is defined as the minimum number of operations (including *Copy/Substitution*, *Insertion* and *Deletion*) required for turning one string to the other [120]. Typically, fixed costs are assigned to each operation respectively in computing the overall cost of a series of editing operations. This formulation assumes that characters are of equal importance and that distance between two characters is binary (either “same” or “different”).

In our problem of comparing video sequences, we adopt an action coding scheme based on the (pose, attributes) couples, which is termed as super-frames. Formally, a super-frame f is defined as a 2-tuple (c, w) which consists of an atom scene c and its attributes w. The set of atom poses form a codebook. Each frame of a video is first assigned a codeword c and then adjacent frames with the same codeword are merged with the attribute w denoting the duration of the same codeword. With this strategy, the coded sequence reflects the local spatial-temporal information through the codewords (which are based on frame differencing) while retaining the global temporal order of the original sequence. This results in a compact yet descriptive representation of the original video clip.

For weighed-sequences of videos, the assumption of “equally important characters” is no longer applicable.

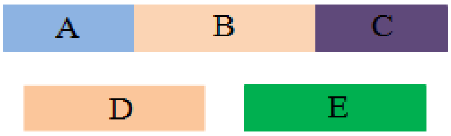


Figure 41: An illustration of distances between two characters.

Firstly, “characters” in a super-frame sequence are codewords with corresponding weights (which may reflect the significance of the codeword). Intuitively, operations on a crucial codeword (e.g., a scene lasting for a longer period) should cost more than on a less important one. For example, deleting a scene of significant length during comparison should result in a large cost for a relative short video.

Secondly, the similarity between the codeword varies and thus in operations, such as *Substitution,* the cost of the operation relies on what to use for the replacement. For example, in Figure 41, we assume that the distance between two characters equals the color difference between the corresponding bars. Obviously, to substitute “B” in the “ABC” sequence, “D” would cost less than using “E”, since the color of “D” is much closer to “B” than “E”. As mentioned earlier, such information is kept in a distance matrix in the codebook creation step and thus we should be able to systematically address such issues.

Specifically, we define a weighted character *a* (e.g., the super-frame *f* in our problem) as a 2-tuple (*c*, *w*) which consists of the label *c* (e.g., the codeword in our super-frame formulation) and its weight *w*. Then a weighted string can be written as

$$s = \{a_1, a_2, ..., a_n\},\ a_i = (c_i, w_i),\ i = 1, ..., n \tag{5.14}$$

where *n* is the number of characters in s . Assume that we have two weighted strings $s^{(1)}$ and $s^{(2)}$:

$$s^{(1)} = \{a_1^{(1)}, a_2^{(1)}, ..., a_{n_1}^{(1)}\},\ a_i^{(1)} = (c_i^{(1)}, w_i^{(1)}),\ i = 1, ..., n_1 \tag{5.15}$$

$$s^{(2)} = \{a_1^{(2)}, a_2^{(2)}, ..., a_{n_2}^{(2)}\},\ a_j^{(2)} = (c_j^{(2)}, w_j^{(2)}),\ j = 1, ..., n_2 \tag{5.16}$$

A $(n_1 + n_2) \times (n_1 + n_2)$ symmetric matrix D_c (with zero elements on the diagonal) records pariwise distances of the vocabulary (range of values in D_c is [0, 1]):

$$\{c_1^{(1)}, c_2^{(1)}, ..., c_{n_1}^{(1)}, c_1^{(2)}, c_2^{(2)}, ..., c_{n_2}^{(2)}\} \tag{5.17}$$

Then the weighted-sequence distance between $s^{(1)}$ and $s^{(2)}$ is defined as the sum of costs caused by operations for turning $s^{(1)}$ to $s^{(2)}$:

$$D^{WSD}(s^{(1)}, s^{(2)}) = \sum_{l=1,...,L} Cost_l \tag{5.18}$$

in which *L* is the number of operations involved; $Cost_l$ denotes the required cost for the lth operation. Three types of editing operations: *Substitution*, *Insertion* and *Deletion* and corresponding costs are defined as follows:

$$\textit{Copy/Substitute}:\ Cost^{(S)} = D_c(c_i^{(1)}, c_j^{(2)}) \tag{5.19}$$

$$\textit{Delete}:\ Cost^{(D)} = w_i^{(1)} D_c(c_{i-1}^{(1)}, c_{i+1}^{(1)}) \tag{5.20}$$

$$Insert\text{: } Cost^{(I)} = w_j^{(2)} \cdot D_c(t_{i-1}^{(1)}, t_j^{(2)}) \tag{5.21}$$

where c is the codeword of a super frame and w is its weight. D_c denotes the matrix, which records pariwise distances of all of the codewords.

With the above definitions, we propose an algorithm as shown in Figure 42 for computing the WSD by extending the conventional Edit Distance algorithm based on dynamic programming. Landau and Vishkin [124] have shown that classical edit distance problem can be solved in $O(mn)$ time using dynamic programming. Since our generalized version does not change the structure of the original algorithm, it still maintains the same computational complexity.

In this work, based on our proposed weighted-sequence distance, we define a WSD kernel function as

$$\exp(-\gamma \cdot D^{WSD}(s^{(1)}, s^{(2)}, D_c)),\ \gamma > 0 \tag{5.22}$$

in which γ is a model parameter. Model parameter γ is selected from an n-fold cross validation on the training set. Since the kernel matrix is not always positive semi-definite, to guarantee a global optimum in SVM, we revise the kernel matrix through shifting all the eigen values by a positive constant [125]. The constant is set as the absolute value of the minimum eigen value in our experiments.

Algorithm: *Weighted-Sequence Distance (WSD)* computes the weighed-sequence distance between two weighted-sequences $s^{(1)}$ and $s^{(2)}$ with given distance matrix D_c.

```
Input: Weighed-sequence $s^{(1)}$ and $s^{(2)}$ and distance matrix $D_c$.
 if $n_1 = 0$
    return $\sum_{j=1,...,n_2} w_j^{(2)}$ ;
end
 if $n_2 = 0$
    return $\sum_{i=1,...,n_1} w_i^{(1)}$ ;
end
Construct an empty matrix $M$ ;
Initial the first row of $M$ as $w_1^1, (w_1^1 + w_2^1), ..., \sum_{i=1,...,n_1} w_i^1$ ;
Initial the first column of $M$ as $w_1^2, (w_1^2 + w_2^2), ..., \sum_{j=1,...,n_2} w_j^2$ ;
while $i \le n_1$ do
    while $j \le n_2$ do
        Compute the following costs respectively:
            $Cost^{copy/substitution} = Cost^S(a_i^{(1)}, a_j^{(2)})$ ;
            $Cost^{insertion} = Cost^I(a_i^{(1)}, a_j^{(2)})$ ;
            $Cost^{deletion} = Cost^D(a_i^{(1)}, a_j^{(2)})$ ;
        Let $M(i+1, j+1) =$
            $\min\{M(i, j+1) + Cost^{insertion},$
                $M(i+1, j) + Cost^{deletion},$
                $M(i, j) + Cost^{copy/substitution}\}$ ;
    end
 end
 $D^{WSD} = M(n_1 + 1, n_2 + 1)$ ;
 return $D^{WSD}$ .
```

Figure 42: Weighed-sequence distance (WSD) algorithm.

5.3.4 Sequence-kernel-based Sparse Representation

Different from the traditional way of using sparse representation, the candidate combination (i.e., a candidate summary y_k) and the target signal (i.e., the original video y) in this application are ordered sequences, so that standard L_2 norm is not applicable to measure the reconstruction error. In Section 5.3.3, we describe WSD kernel metric to measure the distance between two sequences with dramatically different lengths.

Using the above kernel function, the quality of a candidate summary can be computed as a regular L_2 norm in the WSD kernel space and the sparse representation formulation of Eq. (5.2) can be rewritten as

$$\alpha_0 = \arg\min_{\alpha} \| \phi(y) - \phi(A \cdot \alpha) \|^2, \ \|\alpha\|_1 < m \tag{5.23}$$

where $A \cdot \alpha$ is a candidate summary according to Eq. (5.7). Eq. (5.23) can be solved as a kernel based L_1 optimization problem

$$\alpha_0 = \arg\min_{\alpha} \| \phi(y) - \phi(A \cdot \alpha) \|^2 + \lambda \cdot \|\alpha\|_1 \tag{5.24}$$

By rewriting the minimization term in Eq. (5.24), we have

$$\begin{aligned} &\min \| \phi(y) - \phi(A \cdot \alpha) \|^2 + \lambda \cdot \|\alpha\|_1 \\ &= \psi(y) - 2\phi(y) \cdot \phi(A \cdot \alpha) + \phi(A \cdot \alpha) \cdot \phi(A \cdot \alpha) + \lambda \cdot \|\alpha\|_1 \end{aligned} \tag{5.25}$$

Considering only the terms containing α, Eq. (5.25) can be further written as

$$\begin{aligned} &\min \| \phi(y) - \phi(A \cdot \alpha) \|^2 + \lambda \cdot \| \alpha \|_1 \\ &= -2\alpha^T \cdot \kappa(y, A) + \alpha^T \cdot \kappa(A, A) \cdot \alpha + \lambda \cdot \| \alpha \|_1 \end{aligned} \tag{5.26}$$

By substituting $\kappa(\cdot)$ with pre-computed kernel gram matrix $K(y, A)$ and $K(A, A)$, Eq. (5.26) can be solved by kernel based feature-sign search algorithm [103, 104].

5.3.5 Recap of Video Summarization Using the Proposed Approach

For better understanding, we recap the procedure of video summarization using the proposed approach in the below. Given a video clip, we first create a dictionary of subsequences by selecting frames/snippets with top importance scores from clusters of frames/snippets or the original video (discussed in Section 5.3.2). Then we compute the sparse coefficients over the obtained dictionary which gives the optimal approximation of the input video, in which WSD kernel (described in Section 5.3.3) is used to measure the reconstruction errors and kernel-based feature-sign algorithm is utilized to solve the L_1 minimization problem (presented in Section 5.3.4). The final summary video is produced directly from the optimal combination of the dictionary elements based on the obtained sparse coefficients.

5.4. Experiments, Evaluations and Analysis

To verify the effectiveness of the proposed approach, we applied our solution and three other video summarization methods to 71 real consumer video clips. A subjective user evaluation was conducted to compare these results visually as well as quantitatively.

5.4.1 Experimental data and preprocessing steps

A total of 71 videos provided by the authors of [105] were originally selected from 3000+ home videos [108] with resolution 640-by-480 or 320-by-240, frame rates from 24 to 30 and average length of 31 seconds. Contents of the videos range from natural sceneries, trips, sports, and outdoor activities to concerts, weddings, birthday parties, and card games, and therefore span reasonably the space of general consumer videos with each video clip containing a single shot. Note that this assumption is generally true for videos captured using digital or phone cameras, thus removing the need for shot boundary detection. Figure 43 shows thumbnails of sample experimental videos used in this work.

For a given video, we first segment the sequences into frames according to their original frame rates. For spatio-temporal data reduction, extracted frames are further down-sampled to 320-by-240 (if the original resolution is 640-by-480) spatially and temporally with a step of 5 frames (i.e., each group of 5 consecutive frames is defined as a snippet and the first frames are used as representatives for each snippet). Dense SIFT descriptions [97] are then extracted on an image grid (*GridSpacing*=5, *PatchSizes*=8 and 16 in our experiments) for H, S, V color channels of each frame and further quantized to 100 dimensions by k-means clustering on randomly selected samples over the entire video. Histograms of quantized features are then computed over a three-level spatial pyramid to yield a 2100-dimension feature vector for each frame.

Figure 43: Thumbnails of sample videos used for experiments.

In dictionary creation, the weights for combining different scores of semantics were set as [0.25, 0.25, 0.25, 0.25] and the number of non-zero snippets in each dictionary element, *l*, was set as 2, 4, or 6 depending on the length of the original video. The number of clusters in the over-segmentation step is set as Minimum{30, *total_number_of_frames*/5}). For our experimental videos, this number is typically much greater than the true number of the underlying "events", which yields a result of over-segmentation of the original video. The size of dictionary is set as 5000 in our experiments.

The lengths for final summary videos are specified as 20% of the original lengths with lower and upper limits of 6 seconds and 12 seconds respectively (All summary videos are generated with the original frame rate). The appropriate sparsity parameter λ is automatically obtained via a coarse-to-fine search (i.e., *step*=0.1 and 0.01 respectively) with an objective to generate a final summary sequence with a length (i.e., number of frames), which is closest to the expected length. We emphasize that the proposed approach generates the final summary video directly and is adaptive to different compression rates of summaries, which is chosen by the user in real applications. We choose 3 relatively long videos from our experimental data set and generate three summary videos of different lengths (i.e., 10%, 20% and 30% of the original length) to show the flexibility of the proposed approach.

5.4.2 Other methods for Comparisons

As mentioned previously, in most existing work video summaries are generated typically from key frames. In this work, we select two different key frame extraction methods to generate video summaries for comparisons. Given extracted key frames, final summary videos consist of temporal segments centered at each key frame. Camera motion-based key frame extraction approach (MKFE) presented in [105] is used for comparisons (where the numbers of key frames were specified by human judges).

In addition, we implemented a k-means clustering based key frame extraction approach (CKFE), in which the number of clusters are estimated by over-clustering (frames are first over clustered, e.g. with *k*=20 and the number of clusters with at least a certain amount of frames, e.g. 10 frames, is used as the cluster number for the second-round clustering). With obtained clusters, after merging small clusters (e.g., with less than 10 frames) to their neighboring clusters, the frame at the temporal center of all the frames in each cluster are selected as key frames. For generating the final summary video, each key frame is further expanded to a temporal segment centered at the key frame with a length proportional to the size of the cluster. Since results from MKFE are only available for the first 18 videos, as reported in [105], the remaining 53 videos are only compared to CKFE method.

5.4.3 Evaluations and Analysis

To analyze the results qualitatively, we visualize an example in Figure 44. The original video is manually segmented into several “events” with representative scenes (as shown on the left, where each row corresponds to an “event”, including a few representative scenes that shows the gradual “development” of the “event”). Corresponding “events” and their representative scenes, if exist, are then extracted from the summary video generated from our approach, as shown on the right. Comparing the “events” on both sides, the summary video completely covers all the “events” and for most of them, even the “development” of the “event” is well retained.

For quantitative evaluation, subjective user evaluation is widely accepted for evaluating video summarization algorithms. In this work, we recruited 7 volunteers who had experience with capturing and sharing consumer videos and thus are the appropriate judges to evaluate such videos. Each of them evaluated all 71 original videos with two or three different versions of summary videos (with anonymous indexes) and rated the results in terms of how well the summary videos cover the contents of the original videos (from 0 -- extremely poor, to 10 -- extremely complete). Evaluators were free to choose different playback speeds as desired.

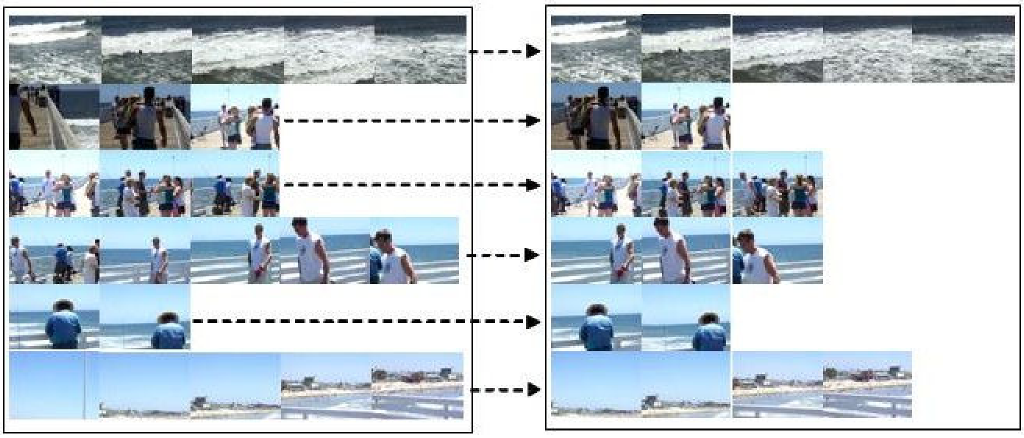
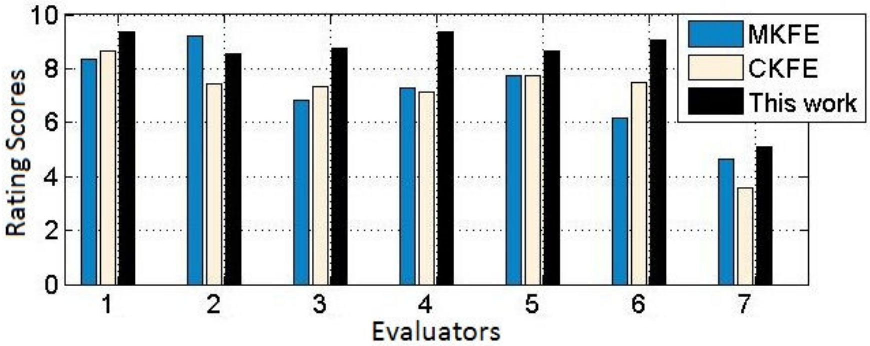
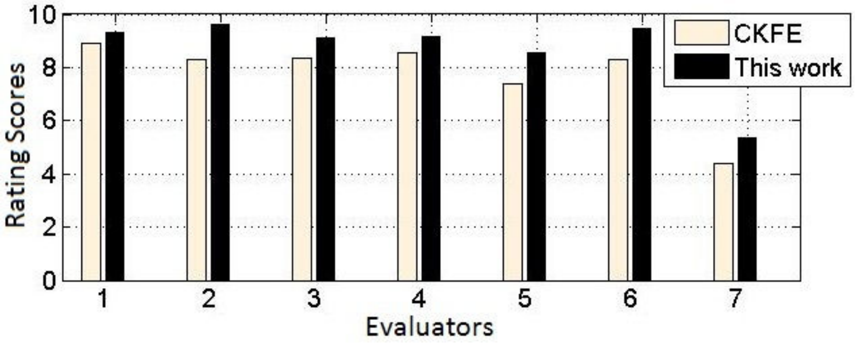


Figure 44: Sample result from the proposed approach: Left—“Events” and representative scenes from the original video (25 seconds); Right—“Events” and representative scenes from the summary video (6 seconds).



(a) Scores from all the evaluators of the three methods averaged over the first 18 videos.



(b) Scores from all the evaluators of the two methods averaged over the remaining 53 videos.

Figure 45: Average rating scores from all evaluators for different methods.

Figure 45 illustrates the bar charts of the average (over the first 18 videos and the remaining 53 videos, respectively) rating scores from all evaluators for different methods. As we can see in the figure, except for the second evaluator in Figure 45-(a), the majority of evaluators in both cases consistently prefer the proposed approach over the other methods, although the absolute rating scores vary to some extent among different evaluators. To measure the statistical significance [116], we set the *null hypothesis* as "the proposed approach is not always better than the other methods for comparisons" and the *test of statistic* is "the number of evaluators rated the proposed approach to be the best". Assuming that the distribution associated with the null hypothesis is a uniform distribution over different methods (i.e., each method has an equal chance to be rated as the best), the critical region are the cases when 6 or 7 (for the 18-video and 53-video cases respectively, as shown in Figure 45) evaluators rated the proposed approach as the best, so the *p*-value equals $(0.33)^6$ or $(0.5)^7$, which is much lower than the conventional *significance level* 5%. Thus, the null hypothesis is rejected. In other words, the observation that the proposed approach is superior to the other methods is *statistically significant* (i.e., it is unlikely to have occurred by chance).

Table 10: Mean rating scores for the first 18 videos.

First 18 videos	MKFE	CKFE	This work
Mean scores	7.17	7.04	**8.42**

Table 11: Mean rating scores for the remaining 53 videos.

Remaining 53 videos	CKFE	This work
Mean scores	7.75	**8.66**

Table 10 and Table 11 present the mean of the average scores (as shown in Figure 45) from different evaluators for different methods. For the first 18 videos, our approach achieves 1.25 (17.4%) and 1.38 (19.6%) improvements over the motion-based approach (MKFE) and the k-means clustering based approach (CKFE), respectively, for the remaining 53 vides (MKFE results are not available), the proposed approach is 0.91 (11.7%) better than CKFE. For a given summary length, there exists a tradeoff between temporal smoothness and content coverage, the latter is of higher priority in this work. Due to this reason, some of the videos in the supplementary material may display rapid shot change as the specified summary lengths for them caused significant temporal compression. Note that MKFE utilized metadata of camera motion in additional to visual contents and the number of key frames was specified by human judges; while our approach is solely based on visual features with no additional knowledge.

Although the proposed approach was designed for unconstrained videos, we expect that it should be also applicable to structured or produced videos (such as news videos) either as a stand-alone or in combination with other existing methods.

5.5. Summary

In this work, we propose a sparse representation based framework for summarizing unconstrained amateur or consumer videos. In contrast to conventional segmentation and key frame based video summarization methods, this approach directly produces the summary of the video without first estimating the key frames. For a wide range of summary lengths, the proposed approach achieves the desired results by naturally varying the sparsity parameter. Using the proposed formulation, subsequences instead of frames are used as dictionary elements while the temporal order is preserved. Furthermore, the proposed approach allows the incorporation of additional criteria such as video quality measures and high-level semantic information into a sparse representation framework that typically only addresses signal reconstruction errors. Extensive experimental results clearly indicate the feasibility of the proposed approach. Future work will focus on exploring additional ways to combine the individual scores for each frame. Also, the proposed approach will be validated on larger scale video datasets.

Chapter 6

CONCLUSIONS AND FUTURE WORK

6.1. Conclusions

This dissertation presents studies on exploring and developing novel computational methods for incorporating contextual information/domain knowledge in different forms for multimedia computing and pattern recognition problems. Specifically, we proposed a novel Bayesian approach with statistical-sampling-based inference for incorporating a special type of domain knowledge, spatial prior for the underlying shapes; We explored cross-modality correlations via Kernel Canonical Correlation Analysis and used the learnt space for associating multimedia contents in different forms; We also modeled contextual information as a graph for regulating interactions among high-level semantic concepts (e.g., category labels), low-level input signal (e.g., spatial/temporal structure). To demonstrate the effectiveness of the proposed approaches, we applied and evaluated them on four real-world applications, including face shape alignment, Flickr photo tag recommendation, YouTube video classification, and general consumer video summarization.

For face shape alignment, we proposed a deformable Bayesian Active Shape Model (BASM) for modeling human faces which integrates anthropometric facial priors with both shape and appearance information learnt from a face dataset under a Bayesian framework. A statistical-sampling-based inference procedure was then introduced under the model for obtaining a data-adaptive version of the model for any given face image. Serving as a starting point, this model enables additional semantic-aware processing steps that were designed to enrich the sketchy face model with more input-specific details, resulting in the final tactile face images. As such, the proposed approach combines anthropometric prior knowledge, learnt model generality and given data specificity to automatically create an informative tactile representation of the original face image.

In the application of photo tag recommendation, we used Kernel Canonical Correlation Analysis (KCCA) for capturing the underlying correlations between image features and text tags of Flickr photo and further used the uncovered relationships for ranking text terms from a dictionary and recommending tags for new photos.

For classifying wild web videos from YouTube, we proposed a novel Tree-DRF fusion framework based on predefined taxonomy structure. Each data source of training samples is first combined with the manually-labeled set independently. Then, built upon a hierarchical taxonomy (i.e., a tree graph) of the categories, tree-DRF fusion strategy is designed for merging models trained from different data combinations. Based on the final fused classifier, category labels are predicted for the new videos.

In addition, we explored a sparse-reconstruction approach which emphasizes on maintaining global temporal order of videos for consumer video summarization. In particular, we proposed a Weighted-Sequence-Distance-kernel (WSD kernel)-based sparse representation approach for directly summarizing consumer videos. A dictionary of subsequences is first constructed from clustered frames with importance ranking scores of extracted high-level semantics. Video summarization is formulated to seek an optimal combination of the dictionary elements that robustly represents the original video. Weighted-Sequence Distance kernel is exploited to compute the approximation error, and the kernel-based feature-sign algorithm is used to estimate the sparse coefficients. A linear combination over the dictionary with the obtained optimal sparse coefficients is output as the final summary video.

For each of the above applications, extensive experiments were carried out on real-world data. Objective/subjective evaluations were performed and compared to state-of-the-art methods. Results from our approaches are comparable or superior to those from existing work, which confirms the effectiveness of the proposed approaches.

6.2. Future Work

We have identified the following aspects that are of interest for future exploration.

Although the proposed BASM with statistical sampling approach was designed for face shape alignment, the idea of incorporating domain knowledge as statistical prior over a predefined structure can be generalized to other types of graphs in different applications.

For tag recommendation, there are many other aspects that can be taken into consideration for further improving the work. For example, other available information of photos, such as title, description, comments, meta-data, etc., can be added as separated features for making tag recommendations. In addition, performing semantic grouping on tags before creating the document-term matrix, combining tag co-occurrence strategies, analyzing users' tagging history and social network/activities for providing customized recommendations are also promising directions.

In Tree-DRF formulation, currently we only considered undirected relationships between parent and child categories in tree-DRF. More sophisticated anisotropic formulations of interaction potential for parent or child neighbors, and siblings may further improve the labeling performance. In addition, it is also possible to make use of unsupervised learning methods (e.g. clustering) for assigning weights to noisy labeled samples and adjusting their contributions accordingly while training classifiers.

For video summarization, future work will focus on exploring additional ways to combine the individual scores for each frame so that the proposed approach can better cover high-level semantics in the video.

In addition, in this dissertation, we assumed that semantic concepts and the mapping between low-level observations and high-level concepts are existing and static. However, emergent and evolutionary aspects [6-10] of semantics are among the fundamental problems of multimedia computing. For example, the vocabulary of tags and the associations between text tags and visual images keep evolving. Modeling the dynamic nature of semantics and making computational algorithms adaptive to the changes are also among our future tasks.

Furthermore, in this work, other than the KCCA approach, we assumed that domain knowledge/contextual information is pre-obtained and the approach used for uncovering the semantic mapping is application dependent. Sophisticated machine learning techniques can be used for extracting useful domain knowledge/contextual information and representing such information in a unified form so that general approaches which leverages domain knowledge/contextual information for uncovering the mapping from low-level signals and high-level semantics can be developed.

Last but not least, for building real multimedia systems, studies on system integration and human-computer interaction (HCI) are also among our future work.

REFERENCES

[1] R.C. Veltkamp and M. Hagendoorn, State-of-the-Art in Shape Matching, Multimedia Search: State of the Art, Springer-Verlag, 2000.

[2] S. Belongie, J. Malik, J. Puzich, "Shape matching and object recognition using shape contexts", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 4: 509-522, 2002.

[3] D. Sharvit, J. Chan, H. Tek, and B.B. Kimia, A Symmetry-Based Indexing of Image Databases, Journal of Visual Communication and Image Representation, vol. 9, no. 4, pp. 366-380, 1998.

[4] Harry Zhang. "The Optimality of Naive Bayes", FLAIRS2004 Conference, 2004.

[5] Olshausen, B., Field, D. "Sparse coding with an overcomplete basis set: A strategy employed by v1?". Vision Research, 1997.

[6] H. Sundaram. "Making sense of meaning: leveraging social processes to understand media semantics". IEEE International Conference on Multimedia and Expo (ICME), 2009.

[7] P. Dourish. "Where the action is: the foundations of embodied interaction". Cambridge, Mass.; London: MIT Press, 2001.

[8] T. Falkowski, J. Bartelheimer, M. Spiliopoulou. "Mining and Visualizing the Evolution of Subgroups in Social Networks". International Conference on Web Intelligence, 2006.

[9] A. Zunjarwad, H. Sundaram, L. Xie. "Contextual Wisdom: Social Relations and Correlations for Multimedia Event Annotation". ACM International Conference on Multimedia (ACM MM), 2007.

[10] Z. Wang, B. Li. "A Bayesian Approach to Automated Creation of Tactile Facial Images", IEEE Transactions on Multimedia, vol. 12, no. 4: 233-246, Jun. 2010.

[11] S. Ina. "Presentation of images for the blind". SIGCAPH Comput. Phys. Handicap., 1996.

[12] T. P. Way and K. E. Barner. "Automatic visual to tactile translation. I. Human factors, access methods and image manipulation". IEEE Transactions on Neural Systems and Rehabilitation, vol. 5, pp. 81-94, 1997.

[13] T. Way and K. Barner. "Visual to Tactile Translation, Part II: Evaluation of the TACTile Image Creation System". IEEE Transactions on Rehabilitation Engineering, vol. 5, 1997.

[14] P. K. Edman, Tactile Graphics, AFB Press, May, 1992.

[15] Tactile Graphics Project: http://tactilegraphics.cs.washington.edu, University of Washington.

[16] The Science Access Project: http://dots.physics.orst.edu, Oregon State University.

[17] R. E. Ladner, M. Y. Ivory, R. Rao, S. Burgstahler, D. Comden, S. Hahn, M. Renzelmann, S. Krisnandi, M. Ramasamy, B. Slabosky, A. Martin, A. Lacenski, S. Olsen, and D. Groce. "Automating Tactile Graphics Translation". ACM SIGACCESS Conference on Assistive Technologies, Baltimore, MD, USA, 2005.

[18] Z. Wang, X. Xu, and B. Li. "Enabling Seamless Access to Digital Graphical Contents for Visually-Impaired Individuals via Semantic-Aware Processing". EURASIP Journal on Image and Video Processing, vol. 2007, pp. 1-14, 2007.

[19] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. "Active shape models - their training and application". Computer Vision and Image Understanding, vol. 61, pp. 38-59, 1995.

[20] Y. Zhou, L. Gu, and H.-J. Zhang. "Bayesian tangent shape model: estimating shape and pose parameters via Bayesian inference". IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2003.

[21] Y. Huang, Q. Liu, and D. Metaxas. "A Component Based Deformable Model for Generalized Face Alignment". IEEE International Conference on Computer Vision (ICCV), 2007.

[22] J. Tu, Z. Zhang, Z. Zeng, and T. Huang. "Face localization via hierarchical CONDENSATION with Fisher boosting feature selection". IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2004.

[23] C. Twining and C. Taylor. "Kernel Principal Component Analysis and the construction of non-linear Active Shape Models". British Machine Vision Conference (BMVC), 2001.

[24] F. D. Torre and M. H. Nguyen. "Parameterized Kernel Principal Component Analysis: Theory and Applications to Supervised and Unsupervised Image Alignment". IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2008.

[25] L. Liang, F. Wen, Y.-Q. Xu, X. Tang, and H.-Y. Shum. "Accurate Face Alignment using Shape Constrained Markov Network". IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2006.

[26] L. Gu and T. Kanade. "A Generative Shape Regularization Model for Robust Face Alignment". European Conference on Computer Vision (ECCV), 2008.

[27] M. Kass, A. Witkins, and D. Terzopoulos. "Snakes: active contour models". International Journal of Computer Vision (IJCV), vol. 1, pp. 321-331, 1988.

[28] T. F. Cootes, G. J. Edwards, and C. J. Taylor. "Active Appearance Models". IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol. 23, 2001.

[29] V. Blanz and T. Vetter. "A Morphable Model for the Synthesis of 3D Faces". ACM SIGGRAPH, 1999.

[30] J. Coughlan and S. Ferreira. "Finding Deformable Shapes Using Loopy Belief Propagation". European Conference on Computer Vision (ECCV), Copenhagen, Denmark, 2002.

[31] X. Liu. “Generic Face Alignment using Boosted Appearance Model”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2007.

[32] H. Wu, X. Liu, and G. Doretto. “Face Alignment via Boosted Ranking Model”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008.

[33] Z. Wang, X. Xu, and B. Li. “Bayesian Tactile Face”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008.

[34] M. B. Stegmann, B. K. Ersboll, and R. Larsen. “FAME-a flexible appearance modeling environment”. IEEE Transactions on Medical Imaging, vol. 22, pp. 1319-1331, 2003.

[35] D. Colbry and G. Stockman. “Canonical Face Depth Map: A Robust 3D Representation for Face Verification”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2007.

[36] L. G. Farkas, Anthropometry of the head and face, 2nd Edition. New York: Raven Press, 1994.

[37] A. Bhattacharyya. “On a measure of divergence between two statistical populations defined by probability distributions”. Bulletin of the Calcutta Mathematical Society, vol. 35, pp. 99-109, 1943.

[38] http://www.viewplus.com/products/touch-audio-learning/IVEO/.

[39] S. Milborrow and F. Nicolls. “Locating Facial Features with an Extended Active Shape Model”. European Conference on Computer Vision (ECCV), 2008.

[40] L. Liang, R. Xiao, T. Wen, and J. Sun, "Face alignment via component-based discriminative search”. European Conference on Computer Vision (ECCV), 2008.

[41] S. Romdhani and T. Vetter. “3D probabilistic feature point model for object detection and recognition”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2007.

[42] A.M. Martinez and R. Benavente. “The AR face database”. CVC Tech. Report #24, 1998.

[43] P.J. Phillips, H. Moon, S.A. Rizvi, P.J. Rauss. “The FERET Evaluation Methodology for Face Recognition Algorithms”. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 22, pp. 1090-1104, 2000.

[44] Y. Mori, H. Takahashi, and R. Oka. “Image-to-word transformation based on dividing and vector quantizing images with words”. International Workshop on Multimedia Intelligent Storage and Retrieval Management, 1999.

[45] T. Kolenda, L. K. Hansen, J. Larsen, and O. Winther. “Independent component analysis for understanding multimedia content” IEEE Workshop on Neural Networks for Signal Processing XII, 2002.

[46] K. Barnard, P. Duygulu, N. d. Freitas, D. Forsyth, D. Blei, and M. I. Jordan. "Matching Words and Pictures". Journal of Machine Learning Research, vol. 3, pp. 1107-1135, 2003.

[47] J. Li and J. Z. Wang. "Real-Time Computerized Annotation of Pictures". at ACM International Conference on Multimedia (ACM MM), 2006.

[48] P. Duygulu, K. Barnard, N. d. Fretias, and D. Forsyth. "Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary". European Conference on Computer Vision (ECCV), 2002.

[49] B. Sigurbjornsson and R. v. Zwol. "Flickr Tag Recommendation based on Collective Knowledge". ACM International Conference on World Wide Web (WWW), 2008.

[50] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. "GroupLens: An open architecture for collaborative filtering of netnews". ACM Conference on Computer Supported Cooperative Work, 1994.

[51] N. D. M. "Implicit Rating and Filtering". DELOS Workshop on Filtering and Collaborative Filtering, 1997.

[52] D. Lewis. "Naïve (Bayes) at Forty: The Independence Assumption in Information Retrieval". European Conference on Machine Learning (ECML), 1998.

[53] S. Deerwester, S. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. "Indexing by latent semantic analysis". Journal of the Society for Information Science, vol. 41, pp. 391-407, 1990.

[54] T. K. Landauer, D. S. McNamara, S. Dennis, and W. Kintsch, Handbook of Latent Semantic Analysis: Psychology Press, 2007.

[55] T. K. Landauer, P. W. Foltz, and D. Laham. "Introduction to Latent Semantic Analysis". Discourse Processes, vol. 25, pp. 259-284, 1998.

[56] D. R. Hardoon, S. R. Szedmak, and J. R. Shawe-taylor. "Canonical correlation analysis: An overview with application to learning methods". Neural Computation, vol. 16, pp. 2639-2664, 2004.

[57] H. Hotelling. "Relations between two sets of variates". Biometrika, vol. 28, pp. 312-377, 1936.

[58] A. Vinokourov, J. Shawe-Taylor, and N. Cristianini. "Inferring a semantic representation of text via cross-language correlation analysis". Annual Conference on Neural Information Processing Systems (NIPS), 2002.

[59] S. Subramanya, Z. Wang, B. Li, and H. Liu. "Completing Missing Views for Multiple Sources of Web Media". International Journal of Data Mining, Modelling and Managment (IJDMMM), vol. 1, no. 1: 23-44, 2008.

[60] N. Agarwal, H. Liu, and J. Zhang. "Blocking objectionable web content by leveraging multiple information sources". SIGKDD Explore News Letter., vol. 8, pp. 17-26, 2006.

[61] J. B. Lovins. “Development of a stemming algorithm”. Mechanical Translation and Computational Linguistics, 11: 22-31, 1968.

[62] Microsoft 2008. “Stopwords and Stoplists”. SQL Server 2008 Books Online: http://technet.microsoft.com/en-us/library/ms142551.aspx.

[63] S. Lazebnik, C. Schmid, J. Ponce. “Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2006.

[64] Z. Wang, B. Li. “Learning to Recommend Tags for On-line Photos”, Second International Workshop on Social Computing, Behavior Modeling, and Prediction (SBP), 2009.

[65] Z. Wang, M. Zhao, Y. Song, S. Kumar, B. Li. “YouTubeCat: Learning to Categorize Wild Web Videos”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010.

[66] M. Belkin, P. Niyogi, and V. Sindhwani. “Manifold regularization: A geometric framework for learning from labeled and unlabeled examples”. Journal of Machine Learning Research, 7:2399–2434, 2006.

[67] A. Blum and T. Mitchell. “Combining labeled and unlabeled data with co-training”. Workshop on Computational Learning Theory, 1998.

[68] C. M. Christoudias, R. Urtasun, A. Kapoor, and T. Darrell. “Co-training with noisy perceptual observations”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[69] O. Duchenne, I. Laptev, J. Sivic, F. Bach, and J. Ponce. “Automatic annotation of human actions in video”. IEEE International Conference on Computer Vision (ICCV), 2009.

[70] M. Everingham, J. Sivic, and A. Zisserman. Hello! my name is... buffy automatic naming of characters in tv video. British Machine Vision Conference (BMVC), 2006.

[71] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. “Liblinear: A library for large linear classification”. Journal of Machine Learning Research, vol. 9, pp. 1871–1874, 2008.

[72] U. Gargi and J. Yagnik. “Solving the label–resolution problem in supervised video content classification”. ACM Multimedia Information Retrieval, 2008.

[73] S. Gupta, J. Kim, K. Grauman, and R. Mooney. “Watch, listen & learn: Co-training on captioned images and videos”. Europe Conference on Machine Learning (ECML), 2008.

[74] M. J. Jones and J. M. Rehg. “Statistical color models with application to skin detection”. International Journal on Computer Vision (IJCV), vol. 46, no.1, pp. 81–96, 2002.

[75] S. Kumar and M. Hebert. “Discriminative fields for modeling spatial dependencies in natural images”. Annual Conference on Neural Information Processing Systems (NIPS), 2003.

[76] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. “Conditional random fields: Probabilistic models for segmenting and labeling sequence data”. International Conference on Machine Learning (ICML), 2001.

[77] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. “Learning realistic human actions from movies”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008.

[78] T. Leung and J. Malik. “Representing and recognizing the visual appearance of materials using three-dimensional textons”. International Journal on Computer Vision (IJCV), vol. 43, no. 1, pp. 29–44, 2001.

[79] J. Liu, J. Luo, and M. Shah. “Recognizing realistic actions from videos”. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[80] D. G. Lowe. “Distinctive image features from scale-invariant keypoints”. International Journal on Computer Vision (IJCV), vol. 60, no. 2, pp.91–110, 2004.

[81] R. E. Neapolitan. Learning Bayesian Networks. Prentice- Hall, Inc., Upper Saddle River, NJ, USA, 2003.

[82] C. Ramachandran, R. Malik, X. Jin, J. Gao, K. Nahrstedt, and J. Han. “Videomule: a consensus learning approach to multi-label classification from noisy user-generated videos”. ACM International Conference on Multimedia (ACM MM), 2009.

[83] H. A. Rowley, Y. Jing, and S. Baluja. “Large scale image based adult-content filtering”. International Conference on Computer Vision Theory and Applications (VISAPP), 2006.

[84] M. E. Sargin, H. Aradhye, P. J. Moreno, and M. Zhao. “Audiovisual celebrity recognition in unconstrained web videos”. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2009.

[85] G. Schindler, L. Zitnick, and M. Brown. “Internet video category recognition”. IEEE CVPR Workshop on Internet Vision, 2008.

[86] A. F. Smeaton, P. Over, and W. Kraaij. “Evaluation campaigns and trecvid”. ACM Workshop on Multimedia Information Retrieval, 2006.

[87] C. G. M. Snoek, M.Worring, and A.W. M. Smeulders. “Early versus late fusion in semantic video analysis”. ACM International Conference on Multimedia (ACM MM), 2005.

[88] P. Viola and M. Jones. “Rapid object detection using a boosted cascade of simple features”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2001.

[89] J. Yang, R. Yan, and A. G. Hauptmann. "Cross-domain video concept detection using adaptive svms". ACM International Conference on Multimedia (ACM MM), 2007.

[90] S. Zanetti, L. Zelnik-Manor, and P. Perona. "A walk through the web's video clips". IEEE CVPR Workshop on Internet Vision, 2008.

[91] H. Zhang, A. Kankanhalli, and S. W. Smoliar. "Automatic partitioning of full-motion video". Multimdedia Systems, vol. 1, no. 1, pp. 10–28, 1993.

[92] X. Zhu. "Semi-supervised learning literature survey". Technical report, University of Wisconsin-Madison, 2008.

[93] X. Zhu and Z. Ghahramani. "Learning from labeled and unlabeled data with label propagation". Technical Report CMUCALD-02-107, Carnegie Mellon University, 2002.

[94] E. Candes, M. Wakin. "An introduction to compressive sampling". IEEE Signal Processing Magazine, vol. 25, pp. 21–30, 2008.

[95] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. "Non-local sparse models for image restoration". IEEE International Conference on Computer Vision (ICCV), 2009.

[96] S. Chen, D. Donoho, M. Saunders. "Atomic decomposition by basis pursuit". Soc. Ind. Appl. Math. Rev., vol. 43, no. 1, pp.129-159, 2001.

[97] S. Lazebnik, C. Schmid and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2006.

[98] T. Cour, S.Yu and J. Shi. "Normalized Cut Clustering Codes". http://www.cis.upenn.edu/~jshi/software.

[99] A. K. Moorthy and A. C. Bovik. "A Two-Step Framework for Constructing Blind Image Quality Indices". IEEE Signal Processing Letters, vol. 17, no. 5, pp. 513-516, 2010.

[100] P. Viola, M. Jones. "Robust Real-Time Face Detection". International Journal on Computer Vision (IJCV), vol. 57, no. 2, pp. 137-154, 2004.

[101] Z. Wang, B. Li. "Human Activity Encoding and Recognition Using Low-level Visual Features". International Joint Conference on Artificial Intelligence (IJCAI), 2009.

[102] V. Roth, J. Laub, M. Kawanabe, and J. M. Buhmann. "Optimal Cluster Preserving Embedding of Nonmetric Proximity Data". IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol. 25, pp. 1540–1551, 2003.

[103] S. Gao, I. Tsang, L. Chia. "Kernel Sparse Representation for Image Classification and Face Recognition". European Conference on Computer Vision (ECCV), 2010.

[104] H. Lee, A. Battle, R. Raina, A. Ng. “Efficient Sparse Coding Algorithms”. Annual Conference on Neural Information Processing Systems (NIPS), 2006.

[105] J. Luo, C. Papin, K. Costello. “Towards Extracting Semantically Meaningful Key Frames From Persona Video Clips: From Humans to Computers”. IEEE Transactions on Circuits and Systems for Video Technology, vol. 19, no. 2, pp. 289-301, 2009.

[106] Z. Rasheed, M. Shah. “Detection and representation of scenes in videos”. IEEE Transactions on Multimedia, vol. 7, no.6, pp. 1097-1105, 2005.

[107] H. Sundaram, L. Xie, S. Chang. “A Utility Framework for the Automatic Generation of Audio-Visual Skims”. ACM International Conference on Multimedia (ACM MM), 2002.

[108] A. Loui, J. Luo, S. Chang, D. Ellis, W. Jiang, L. Kennedy, K. Lee, A. Yanagawa. “Consumer video benchmark data set: concept definition and annotation”. International Workshop on Multimedia Information Retrieval, 2007.

[109] H. Kang, Y. Matsushita, X. Tang and X. Chen. “Space-Time Video Montage”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2006.

[110] A. Gupta, P. Srinivasan, J. Shi and L. Davis. “Understanding Videos, Constructing Plots Learning a Visually Grounded Storyline Model from Annotated Video”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[111] T. Wang, Y. Gao, P. Wang, E. Li, W. Hu, Y. Zhang and J. Yong. “Video Summarization by Redundancy Removing and Content Ranking”. ACM International Conference on Multimedia (ACM MM), 2007.

[112] F. Chen, M. Cooper and J. Adcock. “Video Summarization Preserving Dynamic Content”. ACM International Conference on Multimedia (ACM MM), 2007.

[113] C. Ngo, Y. Ma, H. Zhang. “Automatic Video Summarization by Graph Modeling”. IEEE International Conference on Computer Vision (ICCV), 2003.

[114] S. Lu, I. King and M. Lyu. “Video Summarization by Video Structure Analysis and Graph Optimization”. IEEE International Conference on Multimedia and Expo (ICME), 2004.

[115] D. Simakov, Y. Caspi, E. Shechtman, M. Irani. “Summarizing Visual Data Using Bidirectional Similarity”. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008.

[116] R. A. Fisher. “Statistical Methods for Research Workers”. Edinburgh: Oliver and Boyd, 1925.

[117] B. T. Truong and S. Venkatesh. “Video abstraction: A systematic review and classification”. ACM Transactions on Multimedia Computing, Communications and Applications (TOMCCAP), 2007.

[118] A.G. Money, H. Agius. "Video summarisation: A conceptual framework and survey of the state of the art". Journal of Visual Communication and Image Representation (JVCIR), vol. 19, no. 2, pp. 121-143, 2008.

[119] A. F. Smeaton, P. Over, W. Kraaij. "Evaluation campaigns and TRECVid". ACM International Workshop on Multimedia Information Retrieval, 2006.

[120] V. I. Levenshtein. "Binary Codes Capable of Correcting Deletions, Insertions and Reversals". Soviet Physic Doklady, vol. 10, pp. 707-710, 1966.

[121] M. Crochemore and W. Rytter. Text Algorithms: Oxford University Press, 1994.

[122] N. Cancedda, E. Gaussier, C. Goutte, and J. M. Renders. "Word Sequence Kernels". The Journal of Machine Learning Research, vol. 3, pp. 1059-1082, 2003.

[123] C. Leslie, E. Eskin, A. Cohen, J. Weston, and a. W. S. Nobley. "Mismatch String Kernels for Discriminative Protein Classification". Bioinformatics Advance Access, vol. 20, pp. 467-476, 2004.

[124] G. Landau, and U. Vishkin. "Fast parallel and serial approximate string matching". Journal of Algorithms, vol. 10, 157–169, 1989.

[125] V. Roth, J. Laub, M. Kawanabe, and J. M. Buhmann. "Optimal cluster preserving embedding of nonmetric proximity data". IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol. 25, pp. 1540–1551.

[126] R. Caruana. "Multitask Learning". Journal of Machine Learning - Special issue on inductive transfer archive, vol. 28, no. 1, 1997.

[127] T. Hofmann , L. Cai. Massimiliano Ciaramita. "Learning with taxonomies: Classifying documents and words". Annual Conference on Neural Information Processing Systems (NIPS) Workshop on Syntax, Semantics, and Statistics, 2003.

[128] D. Sheldon. "Graphical Multi-Task Learning". Annual Conference on Neural Information Processing Systems (NIPS), 2008.

[129] T. Evgeniou, M. Pontil. "Regularized Multi–Task Learning", Knowledge Discovery and Data Mining (KDD), 2004.

[130] J. Ye, S. Ji, and J. Chen. "Multi-class Discriminant Kernel Learning via Convex Programming". Journal of Machine Learning Research, vol. pp.719-758, 2008.

[131] J. Chen, J. Liu, and J. Ye. "Learning Incoherent Sparse and Low-Rank Patterns from Multiple Tasks". International Conference on Knowledge Discovery and Data Mining (SIGKDD), 2010.

[132] J. Liu and J. Ye. "Moreau-Yosida Regularization for Grouped Tree Structure Learning". Annual Conference on Neural Information Processing Systems (NIPS), 2010.

[133] T. Tuytelaars and K. Mikolajczyk. "Local Invariant Feature Detectors: A Survey". Foundations and Trends in Computer Graphics and Vision: Vol. 3: No 3, pp. 177-280, 2008.

[134] H. Wang, A. Divakaran, A. Vetro, S. Chang, and H. Sun. “Survey of compressed-domain features used in audio-visual indexing and analysis”. Journal of Visual Communication and Image Representation: Vol. 14: 150-183, 2003.

APPENDIX – RELATED PUBLICATIONS

Journal/Conference Publications:

[1] **Zheshen Wang**, Baoxin Li. "Synchronizing Disparate Video Streams from Laparoscopic Operations in Simulation-based Surgical Training", IEEE Applied Imagery and Pattern Recognition Workshop, Oral Presentation, 2010.

[2] **Zheshen Wang**, Baoxin Li. "A Bayesian Approach to Automated Creation of Tactile Facial Images", IEEE Transactions on Multimedia, vol. 12, no. 4: 233-246, Jun. 2010.

[3] **Zheshen Wang**, Ming Zhao, Yang Song, Sanjiv Kumar, Baoxin Li. "YouTubeCat: Learning to Categorize Wild Web Videos", IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2010.

[4] Devi A. Paladugu, **Zheshen Wang**, Baoxin Li. "On Presenting Audio-Tactile Maps to Visually Impaired Users for Getting Directions", Work-in-Progress program, International ACM Conference on Human Factors in Computing Systems (CHI), 2010.

[5] **Zheshen Wang**, Baoxin Li, Terri Hedgpeth, Teresa Haven. "Instant Tactile-Audio Map: Enabling Access to Digital Maps for People with Visual Impairment", International ACM Conference on Computers and Accessibility (SIGASSETS), full technical paper, 2009.

[6] **Zheshen Wang**, Baoxin Li. "Human Activity Encoding and Recognition Using Low-level Visual Features", International Joint Conference on Artificial Intelligence (IJCAI), Oral Presentation, 2009.

[7] **Zheshen Wang**, Baoxin Li. "Learning to Recommend Tags for On-line Photos", Second International Workshop on Social Computing, Behavior Modeling, and Prediction (SBP), Oral Presentation, 2009.

[8] Shankara Subramanya, **Zheshen Wang**, Baoxin Li and Huan Liu. "Completing Missing Views for Multiple Sources of Web Media", International Journal of Data Mining, Modelling and Management (IJDMMM). vol. 1, no. 1: 23-44, Dec. 2008.

[9] **Zheshen Wang**, Xinyu Xu, Baoxin Li. "Bayesian Tactile Face", IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2008.

[10] **Zheshen Wang**, Baoxin Li. "A Two-stage Approach to Saliency Detection", IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2008.

[11] **Zheshen Wang**, Xinyu Xu, Baoxin Li. "Enabling Seamless Access to Digital Graphical Contents for Visually-Impaired Individuals via Semantic-Aware Processing", EURASIP Journal on Image and Video Processing, special issue on Image and Video Processing for Disability, vol. 2007, Article ID 18019, 14 pages, DOI: 10.1155/2007/18019, 2007.

Technical Demonstrations:

[1] Nan Li, **Zheshen Wang**, Jesus Yuriar, Baoxin Li. “TactileFace: A System for Enabling Access to Face Photos by Visually-impaired People”, International Conference on Intelligent User Interface (IUI), 2011.

[2] **Zheshen Wang**, Baoxin Li, “Enable Interactive Access to Map Images for People Who are Visually-Impaired”, International Technology and Persons with Disabilities Conference (CSUN), 2008.

[3] **Zheshen Wang**, Baoxin Li, “Enable Seamless Access to Electronic Graphical Information for People Who are Visually-Impaired”, International Technology and Persons with Disabilities Conference (CSUN), 2007.

Manuscripts under Review:

[1] **Zheshen Wang**, Mrityunjay Kumar, Jiebo Luo and Baoxin Li. “Sequence-Kernel Based Sparse Representation for Amateur Video Summarization”.

[2] **Zheshen Wang**, Mrityunjay Kumar, Jiebo Luo, Baoxin Li. “Extracting Key Frames from Consumer Videos Using Bi-layer Group Sparsity”.