Query-Driven Approach to Face Clustering and Tagging

Liyan Zhang, Xikui Wang, Dmitri V. Kalashnikov, Sharad Mehrotra, and Deva Ramanan

***Abstract*— In the era of big data, a traditional offline setting to processing image data is simply not tenable. We simply do not have the computational power to process every image with every possible tag; moreover, we will not have the manpower to clean up the potentially noisy results. In this paper, we introduce a query-driven approach to visual tagging, focusing on the application of face tagging and clustering. We integrate active learning with query-driven probabilistic databases. Rather than asking a user to provide manual labels so as to minimize the uncertainty of labels (face tags) across the entire data set, we ask the user to provide labels that minimize the uncertainty of his/her query result (e.g., "How many times did Bob and Jim appear together?"). We use a data-driven Gaussian process model of facial appearance to write the probabilistic estimates of facial identity into a probabilistic database, which can then support inference through query answering. Importantly, the database is augmented with contextual constraints (faces in the same image cannot be the same identity, while faces in the same track must be identical). Experiments on the real-world photo collections demonstrate the effectiveness of the proposed method.**

***Index Terms*— Face tagging, query-driven active learning, contextual constraints.**

I. Introduction

THE era of "big data" is certainly upon us. Recent estimates suggest that by 2018, more than 80% of total traffic on the internet will consist of video transmissions, and more than 50% of total traffic will originate from non-PCs (e.g., mobile) devices [1]. These statistics suggest that big multimedia data from heterogeneous sources will play a vital role in the future, supporting a variety of end applications. We see three main challenges for visual processing in the big-data era:

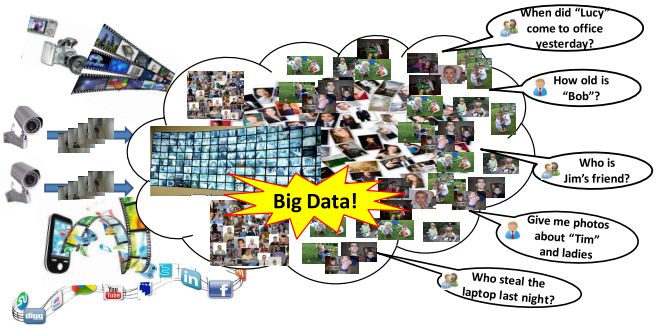


Fig. 1. In the era of big data, the "offline" setting for face clustering/tagging is not suitable. We propose query-driven paradigm to face clustering/tagging which can be seamlessly integrated into image analysis/retrieval process.

1) **Variability:** Diverse data sources mean that the data will arrive in a noisy, uncertain state, making data-cleaning a fundamental issue [2].
2) **Velocity:** Data and applications will appear at fast, essentially real-time rates [1]. Such fast arriving data cannot be processed fully when it arrives. Also, we may not know which data needs to be processed and how it should be processed until we know the applications.
3) **Volume:** We simply will not have the computational resources to process all the data with all possibly relevant attribute tags. It will be crucial to not spend computation on processing that will not be used by a future application.

To appreciate how big data impacts vision techniques, we focus on a concrete setting of face-tagging and clustering. Imagine a user wishing to analyze a large video data source he/she found over an internet using a search engine, or a large collection of surveillance video from a camera array in an office building. A user wishes to perform face-tagging/clustering to support analysis such as "What time does Bob usually come to office?" or "People from which group use the common room facilities the most?" (as illustrated in Fig. 1). Face tagging and clustering techniques have been widely explored and incorporated in many commercial photo management systems such as Google's Picasa and Apple iPhoto. Most such systems offer semi-automated techniques – the system performs an initial clustering based on various features, the result of which are returned to the users for cluster refinement and tagging. The academic community has also pursued similar methods for interactive or "human-in-the-loop" face tagging [3]–[6], sometimes addressed in an active-learning framework [7], [8]. Notably, such methods, though

Manuscript received March 9, 2016; revised June 13, 2016; accepted July 4, 2016. Date of publication July 18, 2016; date of current version August 5, 2016. This work was supported in part by the National Science Foundation under Grant 1118114, Grant 1059436, Grant 1063596, Grant 1527536, and Grant 1545071, in part by the National Science Foundation of China Grant 61572252, and in part by the National Science Foundation of Jiangsu Province Grant BK20150755 and Grant BK20150754. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xiaochun Cao.

L. Zhang is with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China, and also with the Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing 210023, China (e-mail: zhangliyan@nuaa.edu.cn).

X. Wang and S. Mehrotra are with the Department of Computer Science, University of California at Irvine, Irvine, CA 92697 USA (e-mail: xikuiw@uci.edu; sharad@ics.uci.edu).

D. V. Kalashnikov is with AT&T Labs Research, Middletown, NJ 07748 USA (e-mail: dmitri.vk@gmail.com).

D. Ramanan is with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: deva.ramanan@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2592703

interactive, are still applied in an "offline" setting assuming one has access to all the data and all tags of interest. The "offline" setting is simply not tenable in the context of big multimedia data, since we do not have the computational power to process every image with every possible tag, and moreover, we will not have the manpower to clean up all the potentially noisy results.

Our Approach: To address the "big data" challenge, a new paradigm is emerging in the world of big data analysis which can be referred to as "just in time analysis" or query-time analysis where we do not process the entire data set [9]. Instead, data is processed in the context of an application which limits expensive analysis to only the part of data that is needed for analysis. This is a huge savings, especially in the interactive face tagging setting that requires human input where the speed and volume of the data prevents tagging the entire data set. In this paper, we explore such a new "query-driven" paradigm in the context of interactive face tagging/clustering. To implement this strategy, we introduce a notion of query-driven active learning. Rather than asking a human to provide a label that minimizes label uncertainty over all the data, *our system asks for a label that reduces the uncertainty in answers for particular query*. For example, if a user queries the system for images of "John", our system will likely interactively prompt the user to provide tags for John or face that get confused with John. We show that this produces much different results than "query-agnostic" prompts used in active learning.

Database-Perspective: Since our approach is based on exploiting the application context for vision processing, we write the paper from a database perspective. Database systems are amongst the most widely used technology for data analysis since they support powerful data models that can be used to represent diverse information/relationships about objects/events, a powerful query language like SQL that can be used to express analysis tasks, and a sleuth of implementation techniques for efficient query processing. We refer the reader to established texts on databases [10] for details about the technology though we will explicitly explain the all the concepts (viz., probabilistic relational databases and entity-relationship schema) we use in our solution.

The rest of this paper is organized as follows. We introduce the related work in Section 2, and define the schema for our database through entity relational diagrams in Section 3. In Section 4, we present the main approach to query-driven face tagging, beginning with probabilistic queries on the media data and then exploring strategies for interactive face tagging. The proposed approach is empirically evaluated in Section 5, specifically compared with previous approaches to query-agnostic active learning. Finally, we conclude in Section 6 by highlighting key points of our work.

II. Related Work

Face tagging and clustering techniques have been extensively explored in the prior literature [11]–[15]. Typically, the primary goal is to guide users to tag faces as quickly and accurately as possible. Then the key question becomes which faces should be tagged first in order to maximize performance on all the remaining data [7]. To address this issue, active learning framework [16]–[19] has been widely utilized to design the sample selection criteria. For example, Siddiquie and Gupta [20] present an active learning framework to simultaneously learn appearance and contextual models for multi-class classification. The work [18] incorporates the semantic gap measure into the information-minimization-based sample selection strategy, and can effectively reduce semantic gap. Kapoor *et al.* [7] have proposed a Gaussian process based active learning paradigm which incorporates constraints as a prior to guide users to tag the faces with maximum expected informativeness.

These techniques, though interactive, are still applied in an "offline" setting (which assumes one has access to all the data and all tags of interest, and ignores the further analysis/retrieval applications). In the context of big multimedia data, the "offline" setting is simply not tenable. Consider, for instance, a continuous feed of images collected by diverse cameras installed all over the public facility such as an airport (or a shopping mall). Given the enormity of such data, its continuous nature, and the number of individual faces that may appear over a period of time, it is infeasible (if not impossible) to comprehensively tag the entire collection of faces. Even if we somehow could spend immense computational and human resources for such a purpose, all such efforts would be of wasted if, for instance, later retrieval / analyses does not require those annotation results. Therefore, in this paper, we propose a new "query-driven" paradigm to face clustering/tagging which can be seamlessly integrated into image analysis/retrieval process.

While the query-driven approach is attractive, it opens a whole set of new challenges, such as query processing on uncertain data, and "query-driven" active learning framework. The database community has widely explored the probabilistic query processing problem, and developed many advanced techniques including fuzzy-logic based approaches [21], logic based approaches, approaches based on Dempster–Shafer theory [22] and approaches based on probability theory [23], [24]. One of the successful approaches proposed by Sen *et al.* [23] describe a relational database encoding of factor graphs that can leverage probability inference techniques to compute query results. Therefore, to process probabilistic queries, we will handle this problem from a database perspective. And then we propose a novel "query-driven" active learning strategy (described in Section 4) to select questions that can reduce uncertainties in answers for this particular query.

III. Schema Definition

We begin by describing our database schema. Suppose that we are given a human-centered photo album that contains M images $\mathcal{I} = \{I_1, I_2, \ldots, I_M\}$, see Figure 2. Assume that n faces are detected, $\mathcal{F} = \{f_1, f_2, \ldots, f_n\}$, with each face denoted as f_i or $f_i^{I_k}$ (that is, f_i is extracted from image I_k). Suppose that middle-level semantic concepts can be extracted from images and faces leveraging the pre-trained classifiers or provided by users, such as image captured time denoted as $I_k.time$, Geo-location $I_k.loc$, and images tags $I_k.tag$. Besides, face attributes can be extracted from each face

TABLE I
SUMMARY OF NOTATIONS USED IN THE PAPER

Notation	Meaning
$\mathcal{I}$	the set of images $\mathcal{I} = \{I_1, I_2, \ldots, I_M\}$
$\mathcal{F}$	the set of detected faces $\mathcal{F} = \{f_1, f_2, \ldots, f_n\}$
$\mathcal{C}$	the set of initial face clusters $\mathcal{C} = \{C_1, C_2, \ldots, C_N\}$
F_L	the set of labeled faces $F_L = \{f_1, f_2, \ldots f_L\}$
T_L	the set of provided face tags $T_L = \{t_1, t_2, \ldots t_L\}$
$\mathcal{R}$	given a query, the set of result nodes $\mathcal{R} = \{r_1, r_2, \ldots r_m\}$
$H(f_i)$	the entropy of face f_i
$I(f_i)$	the information gain of face f_i
$H^q(f_i)$	the query-driven entropy of face f_i
$\widetilde{H^q}(f_i)$	the query-driven entropy with constraints for face f_i

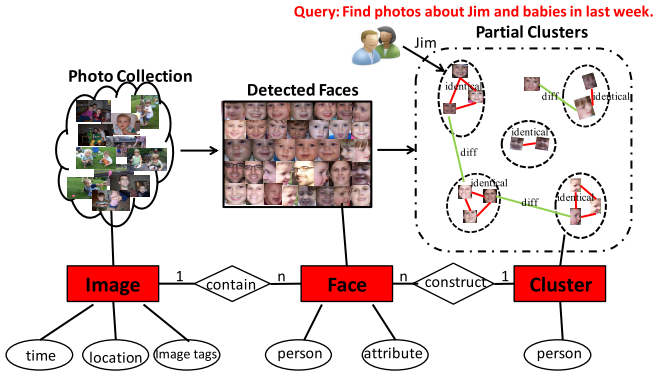


Fig. 2. The input of query-driven face tagging problem. We visualize our database schema with a standard entity relationship diagram [10] consisting of entities (squares), attributes (ovals), and relations (diamonds). Such a schema can be efficiently implemented in a relational database.

utilizing the pre-trained attribute extraction classifiers, including intrinsic attributes like "gender", "age", "ethnicity", and describable attributes "black hair", "big nose", etc., denoted as $f_i.attr$. Each face is associated with an identity attribute $f_i.t$. Importantly, the domain of $f_i.t$ is unknown (because of our open-world assumption). Table I summarizes the meaning of notations used throughout this paper.

Database Constraints: Contextual constraints provide additional cues about the face identities. For example, co-occurring faces in one image usually refer to different people. Such a relationship can be defined as a "diff" constraint ε^- in our schema. Faces from the same face track in a video should refer to the same person, denoted as a "identical" (or "same") constraint ε^+. In our experiments, we generate an over-clustering of faces (with a tight threshold on appearance variation) to construct a set of N initial clusters $\mathcal{C} = \{C_1, C_2, \ldots, C_N\}$. Assuming that each cluster is pure, we enforce the same ε^+ constraint for each cluster.

Weak Supervision: We construct a probabilistic database using the entity-relationship diagram in Figure 2. On top of this probabilistic database, users can present a query to extract the interested knowledge. To indicate the target people in the query, users can specify several face samples of this target person. Therefore, the whole face dataset $\mathcal{F}$ will be partitioned into labeled face set $F_L = \{f_1, f_2, \ldots f_L\}$, with tags $T_L = \{t_1, t_2, \ldots t_L\}$, and unlabeled face set F_U. Our goal is to choose

Fig. 3. User Interface. Given a query, the system will automatically choose some questions to return to users for feedbacks, based on which the query answers will be updated.

which faces to tag (or questions to ask users) in order to achieve the accurate query answers as soon as possible.

IV. QUERY-DRIVEN APPROACH TO FACE TAGGING

Querying: One advantage of a database is immediate support for powerful operators such as *selections* "show me all male faces", *complex selections* such as (e.g., "male faces with Jack") , aggregations ("who appeared most often with Jim"), and *joins* ("all pictures of a person who appeared with Jim"). SQL allows one to algebraically compose fairly complex queries by composing the basic operators such as selections and joins. As we will show, a query language is fundamental to the success of our active-learning approach because it allows us to focus user effort on labeling data that matters (i.e., relevant for the user's query).

Figure 3 illustrates the designed interface for an example complex selection query. Query answers are displayed to users as a ranking list based on the relevance to queries. To measure the quality of query answers, we score the average precision (AP) [25] of a returned ranked list. Our goal is to improve the AP performance of the returned ranking list with minimum user effort.

Overall Framework: We visualize our overall framework in Fig. 4. Given a media dataset, we first built a probabilistic database by extracting semantic attributes using visual concept detectors (for example, tuned for faces and particular face attributes). We expect these detectors (and the resulting attributes) to be rather noisy and probabilistically uncertain. When users present a high-level semantic query (translated to SQL query algebra), the database manager will process the query and return an answer to the query. If users are not satisfied with the answer, the human-in-the-loop component will be activated. It will automatically generate questions to ask users for feedback, based on which the final query answer will be updated until users are satisfied.

A. Probabilistic Query Processing

Capturing probabilistic uncertainties is still somewhat challenging in a database model. One of more successful approaches implements a probabilistic graphical model (PGM)

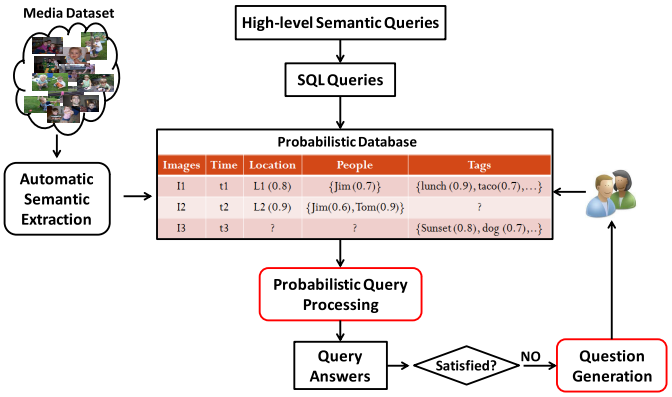


Fig. 4. The general framework of query-driven face tagging. Queries are processed on a probabilistic database to generate the tentative answers. And then the query-driven active learning strategy will select questions to return to users for feedback, which will be leveraged to update the query answers.

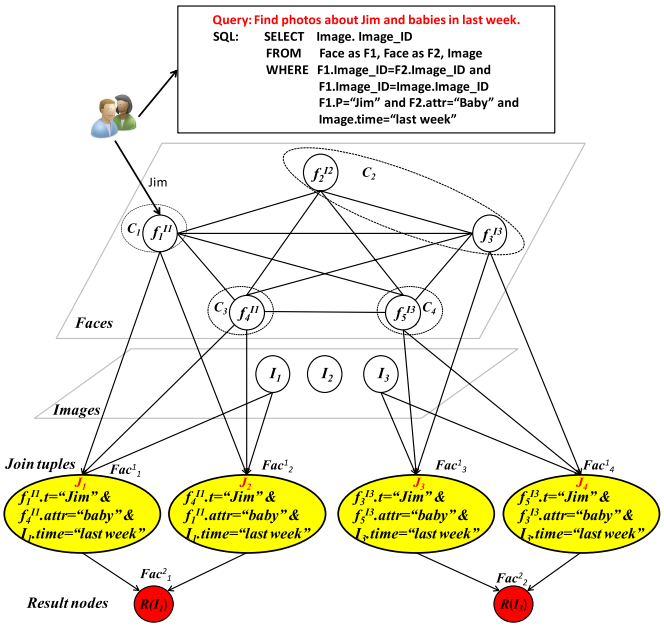


Fig. 5. An Example of Factor Graph. A given query is first transformed into SQL query, and then represented by factor graph. Correlations and dependencies are captured by factor graph to facilitate the probabilistic query reasoning.

through a database system [23]. Sen et al. describe a relational database encoding of factor graphs that can leverage probability inference techniques to compute query results. Importantly, users can produce queries using a standard set algebra including selection, join, aggregation, etc. Sen et al. describe a mapping for transforming such queries into a factor graph. In this paper, we choose to focus on "select" queries – e.g., "find the photos of Jim and babies in the last week". There are several reasons – first, before we explore more complex nature of queries – e.g., joins, aggregations, or a full class of SQL, it makes sense to develop and validate the idea in the context of selections. Furthermore, selections, while limited in their degree of expressibility are arguably the most important class of queries in the context of visual data collections. Finally, as we will discuss later in the paper, techniques for more complex queries can be built around the techniques we develop for selections.

1) Factor Graph Representation: We visualize a small example factor-graph in Figure 5. Note that this represents a probabilistic factor graph with random variables, and not an entity-relationship diagram. Specifically, the factor graph includes random variables capturing the uncertain identity label of each face, and possibly uncertain tags associated with each image. The factor-graph encoding enumerates combinations of entity relationships with a "join operation", and associates each join-tuple with a binary random variable indicating if it is true or not [23].

Dependencies: The edges in the graph can represent the correlations and dependencies between entities. Correlations are represented with undirected links. We use such edges to model appearance similarities between face entities. Directed edges represent relationship dependencies between entities. For example, join tuples depend on the faces and images, where the dependence is defined by factor functions Fac_i^1. These factor functions essentially act as "ANDs", e.g., $Fac_1^1(J_1|f_1^{I_1}, f_4^{I_1}, I_1) = true$, if $(f_1^{I_1}.t = \text{"}Jim\text{"} \wedge f_1^{I_4}.attr = \text{"}baby\text{"} \wedge I_1.time = \text{"}last\ week\text{"})$. Thus, the probability of join tuple $J_1 = true$ is the joint probability of each of its dependent factors taking on the given states.

Result Nodes: We define a result node associated with each image in the database, capturing a binary variable that specifies if it should be returned in the query answer. Result nodes rely on join tuple nodes, with the factor denoted as Fac_i^2, where we use the superscript 2 to denote that this factor function is distinct from the one used to compute join tuple probabilities. Result-specific factor functions essentially act as "ORs", returning $Fac_1^2(R(I_1)|J_1, J_2) = 1$, if $(J_1 = true \vee J_2 = true)$. We write r_i as the final computed probability that image i should be returned in the result.

2) Inference on Unlabeled Data: The previous factor-graph model crucially requires accurate posterior probabilities that an unlabeled face will take on a particular label – e.g., $P(f_i.t = \text{"}Jim\text{"})$. We now describe an appearance model for predicting such labels given a smaller set of tagged faces. Our framework can be applied with any local appearance model that produces posterior class distributions, including class-specific Gaussian, softmax classifiers, calibrated SVMs, etc. Given a database-perspective, it is natural to use data-driven nonparametric appearance models to make predictions on unlabeled data. Gaussian processes elegantly combine the flexibility of data-driven nearest-neighbor methods with the regularization afforded through smooth parametric functional models.

For simplicity, let us write the person identity attribute of a face f_i as $t_i = f_i.t$. Assume that the user has provided a collection of face tags. Based on "same/diff" constraints, the database manager can spread these constraints to resolve the label of some other faces. This will partition the entity set of faces F into a labeled face set $F_L = \{f_1, f_2, \ldots f_L\}$, with tags $T_L = \{t_1, t_2, \ldots t_L\}$, and unlabeled face set F_U. Our goal is to infer the posterior probability $p(t_u|F, T_L)$, given an unobserved face $f_u \in F_U$. We can then update the probabilistic

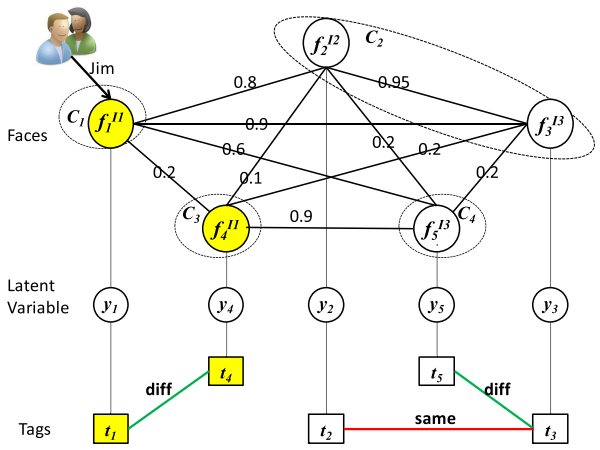


Fig. 6. Gaussian process model is leveraged to infer the "local" probability of unlabeled samples.

database with revised identity attribute. For example, in Fig. 6, with face $f_1^{I_1}$ labeling as "Jim", face $f_4^{I_1}$ will be automatically labeled as "not Jim" according to the "diff" constraint, and thus the labeled face set turns to be $F_L = \{f_1^{I_1}, f_4^{I_1}\}$ with labels $T_L = \{1, -1\}$. We aim to induce the probability of unlabeled faces $(f_2^{I_2}, f_3^{I_3}, f_5^{I_3})$ referring to "Jim".

Kapoor *et al.* [7] use a Gaussian Process (GP) prior with contextual constraints to predict posterior distribution of tag labels over a set of unlabeled faces. We use a simplified form of their model, using the GP to only produce "local" predicts of face identity. We make use of the probabilistic database manager to enforce contextual constraints.

GP Classification: Gaussian process models have been widely explored in active learning, especially for visual classification. To use GPs for classification, we first introduce a latent variable vector $Y = \{y_1, y_2, \ldots, y_n\}$, where n is the number of labeled entities (faces). The discrete class label t_i for a face f_i is generated via the continuous latent variable y_i. Latent variables capture the assumption that similar faces should share similar predictions. The posterior distribution $p(Y|F)$ can be formally denoted as $p(Y|F) \sim N(0, \mathcal{K})$, where $\mathcal{K}$ refers to a kernel matrix with $\mathcal{K}_{ij} = k(f_i, f_j)$, which encodes the similarity between face pairs. In the experiments, $\mathcal{K}_{ij} = exp(-\frac{D_{ij}}{mean(D)})$, where D_{ij} refers to the distances between face pair f_i and f_j based on the face representations (Note that, this kernel function is positive semi-definite). Following standard GP constructions [26]–[28], given an unlabeled face f_u, the posterior probability over latent variable y_u has a simple form, $p(y_u|F, t_L) \sim N(\overline{y}_u, \sigma_u^2)$, where,

$$\overline{y}_u = k_u^T (K_{LL} + \sigma^2 I)^{-1} T_L \quad (1)$$

$$\sigma_u^2 = k_{uu} + \sigma^2 - k_u^T (K_{LL} + \sigma^2 I)^{-1} k_u \quad (2)$$

Here, the notation K_{LL}, k_u, k_{uu} respectively refer to the kernel matrix containing covariance between training samples, the kernel vector consisting of covariance between training samples and unlabeled samples, and the covariance of the test sample to itself. The notation I refers to the identity matrix (square matrix with ones on the main diagonal and zeros elsewhere). The class label t_u is given by the sign of predicted mean $\overline{y}_u$.

Constraints: Using the above model by itself only produces local predictions for each unlabeled face. These may not be consistent with contextual "same" and "diff" constraints encoded in our database. For example, in Figure 6 the probability of t_2 and t_3 referring to "Jim" should be equal, while t_3 and t_5 can not refer to "Jim" simultaneously due to the "diff" constraint. To enforce these constraints, we simply request the probabilistic database manager to return a valid query. Internally, the manager is solving an interference problem using standard algorithmic techniques such as sampling or belief propagation [23].

B. Query-Driven Active Learning

We now discuss the "heart" of our approach – an active learning framework for generating questions for user feedback that is designed to provide better answers to the user's query (rather than reduce uncertainty across the whole dataset). We will focus on questions of face identity. Specifically, our system asks the user to label a previously unlabeled face so as to provide better query answers.

1) Exhaustive Approach: Many standard active learning methods [7], [8], [28], [29] choose uncertainty or information gain as the criteria to select which sample to label. The uncertainty criterion seeks to choose the face sample with most uncertainty, whereas the information gain criterion seeks to select a data point that has the highest expected reduction in uncertainty over all the other unlabeled points [7].

Entropy [29] is the most common way to measure uncertainty. Given a discrete random variable X, entropy is defined as $H(X) = -\sum_x p(X = x) log p(X = x)$, where x is the class that X might refer to.

Information Gain: Suppose that given a specific query, we obtain the current query result node distributions $\mathcal{R} = \{r_1, r_2, \ldots r_m\}$ (m is the total number of result node). In our case, the information gain of an unlabeled face f_i can be defined as the uncertainty reduction of total query result set associated with labeling face sample f_i:

$$I(f_i) = H(\mathcal{R}) - H^e(\mathcal{R}|f_i) \quad (3)$$

Here, $H(\mathcal{R}) = \sum_{j=1}^m H(r_j)$, measures the uncertainty of total query answer set before f_i is resolved; $H^e(\mathcal{R}|f_i) = \sum_{j=1}^m \sum_t p(f_i = t) H(r_j|f_i = t)$, refers to the expected total uncertainty of the new query answer set with f_i resolved (t is the possible label of f_i). Our goal is to choose the unlabeled face with the maximum information gain value, $f^* = \arg\max_{i \in U} I(f_i)$.

However, this algorithm is very expensive to compute because it requires us to perform repeated inference by considering all possible labels for each unlabeled face sample, and the resulting impact each labeling would have on the query answer set. Thus we need to explore the efficient approach to estimate information gain without repeating the probability inference.

2) Efficient Approach: Query-Agnostic Entropy: In the query agnostic scenario, as most prior face tagging work suggested [7], [29], the information gain of a face sample f_i can be estimated by its own uncertainty (entropy) $H(f_i)$.

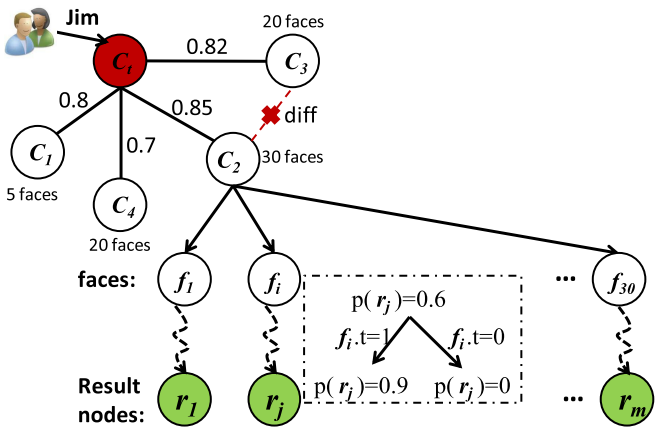


Fig. 7. Query-driven entropy is introduced to measure the uncertainty of query answers induced by the uncertain samples. For instance, if f_i is resolved, we compute uncertainty reduction of result node r_j which depends on f_i.

However, in our case, we aim to seek the samples which can maximally reduce the uncertainty of query answers. Therefore, we introduce the concept of query-driven entropy.

Query-Driven Entropy: As illustrated in Figure 7, given a query and the corresponding result nodes $\{r_j\}_{j=1}^{m}$, we can discover the dependence between the face node and result node. For each face f_i, we can record the result node r_j dependent on f_i. This dependence is represented as $f_i \rightsquigarrow r_j$, meaning that the labeling of face f_i will lead to the update of result node r_j. Then we define the query-driven entropy of face f_i as the expected entropy of query node r_j with f_i resolved, denoted as,

$$H^e(r_j|f_i) = \sum_t p(f_i = t) H(r_j|f_i = t) \tag{4}$$

Then the query-driven entropy of face f_i can be estimated as the uncertainty reduction of result node r_j (where $f_i \rightsquigarrow r_j$) with f_i resolved, defined as,

$$H^q(f_i) = \sum_{f_i \rightsquigarrow r_j} H(r_j) - H^e(r_j|f_i) \tag{5}$$

Query-Driven Entropy With Constraints: With further observation, we discover that constraint is an important factor that can not be omitted, because the identity of one face will impact the other with constraints. For instance, as illustrated in Figure 7, "diff" constraints exist between C_2 and C_3, therefore, if one face in C_2 is resolved to be true, then faces in C_3 will be false automatically. Thus the query-driven entropy should be defined considering constraints.

$$\widetilde{H^q}(f_i) = H^q(f_i) + p(f_i = ture) \sum_{(i,k)\in\varepsilon^-} H^q(f_k) \tag{6}$$

where face f_k refers to the faces with "diff" constraints to f_i. This equation means that the query-driven entropy of face f_i should also include the entropy of constraint face f_k, if f_i resolved to be true.

Selected Face Cluster: In our scenario, partial clustering process is performed to cluster faces into initial groups where faces in the same group are assumed to refer to the same entity, so we can choose the returned samples in the group level. For each group, one face is returned to users to represent the whole cluster. Thus we define the query-driven entropy of a cluster as $\widetilde{H^q}(C_k) = \sum_{f_i \in C_k} \widetilde{H^q}(f_i)$. Using this criterion, we can choose the unlabeled cluster C^* returned to users for tagging, denoted as

$$C^* = \arg\max_{k \in U} \widetilde{H^q}(C_k). \tag{7}$$

Algorithm 1 Greedy Algorithm for K Questions

input : Unlabeled set $\mathbb{S}$; Parameter K
output: Selected set Ω, where $|\Omega| = K$

$\Omega \leftarrow \emptyset$
$Q \leftarrow \emptyset$
foreach *node* $C_i \in \mathbb{S}$ **do**
  Compute $\widetilde{H^q}(C_i)$
  $Q.insert(C_i, \widetilde{H^q}(C_i))$
while $|\Omega| < K$ *and* $|Q| > 0$ **do**
  $C_{top} \leftarrow Q.pop()$
  $\mathcal{C}_\Omega \leftarrow \{C_j : C_j \in \Omega \wedge (C_{top}, C_j) \in \varepsilon^-\}$
  if $\mathcal{C}_\Omega = \emptyset$ **then**
    $\Omega \leftarrow \Omega \cup \{C_{top}\}$
    $\mathbb{S}_{cand} \leftarrow \mathbb{S}_{cand} \setminus \{C_{top}\}$
  else
    Update $\widetilde{H^q}(C_{top})$
    $Q.insert(C_{top}, \widetilde{H^q}(C_{top}))$
return Ω

3) Select K Questions in a Batch: So far, we have discussed the criteria to choose samples for feedbacks. However, in the real applications, it is inefficient to return only 1 question to users in each iteration. Therefore, next we will explore the strategy to choose K questions in each iteration, described in Algorithm 1. At the first thought, the samples ranked in the top K list can be returned to users. However, with further consideration, we discover that constraints make this problem a little more complex. For instance, as illustrated in Figure 7, suppose that we set $K = 2$, and C_2 and C_3 rank in the top 2 list, it is not wise to return them together, since the resolve of C_2 might lead to the resolve of C_3. Therefore, once we choose C_2, the impact of C_3 should be updated to $(1-p(C_2))\widetilde{H^q}(C_3))$ (C_3 is needed to resolve only C_2 is false). Based on this consideration, we propose the greedy algorithm to choose K questions.

V. Experiments and Results

We conduct experiments on four data collections: Pubfig, Gallagher, Wedding and Surveillance. The PubFig [30] database is a large, real-world face dataset consisting of 58,797 images of 200 people collected from the internet. These images are taken in completely uncontrolled situations with non-cooperative subjects. Gallagher and Chen [31] is a public family album capturing the daily life of a family, containing three children, their parents and friends, about 37 people with

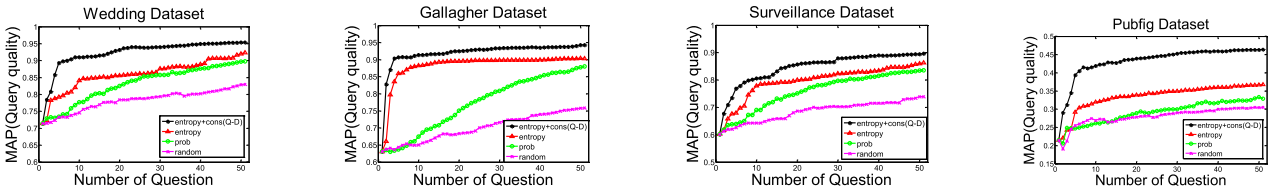


Fig. 8. Comparison of different strategies on four benchmark datasets.

a total of 1064 faces in 591 images. The wedding dataset downloaded from Web Picasa, captures people in a wedding ceremony, including the bride, the groom, their relatives and friends, containing 31 people with a total 1433 faces in 643 images. The surveillance dataset contains images that capture the daily life of faculty and students in the 2nd floor of a computer science building. There are 45 people appearing in 1030 images, but with only 70 faces detected due to the low image quality.

Image Pre-Processing: Before performing experiments, we extract mid-level semantic features including timestamps, geo-location (from EXIF data), face detections, and face attributes. We use Google Picasa [32] to generate face detections. After resizing each detection to a canonical size, we extract Local Binary Patterns (LBP) [33] features as well as attribute features using the online system of [34], which returns 73 attributes (and associated probabilities) covering various properties such as "gender", "age", "big nose", etc. We concatenate our appearance features, apply PCA to reduce dimensionality, and measure pairwise similarity with Euclidean distances. We feed this pairwise similarity into an off-the-shelf clustering algorithm [35] with a very conservative threshold to generate an "over-clustering" of faces.

A. Results and Analysis

In the experiments, we mainly concentrate on the person-related "selection" queries. To simulate user behaviors, we assume that users will specify the target person by labeling one or several groups of faces, and present queries with specific conditions (e.g., find images about Jim and young lady together). Our framework is able to answer the query in the active learning paradigm. Query answer is returned to users in a form of ranking list where each result node is sorted based on the relevance probability $p(r_i)$. Average precision has been widely employed to measure the quality of information retrieval task due to its good discrimination and stability. Therefore, we propose to leverage it as the metric to evaluate the quality of query answer. It is the average of the precision value obtained for the set of top k samples existing after each relevant sample is retrieved, and this value is then averaged over information needs, defined as follows.

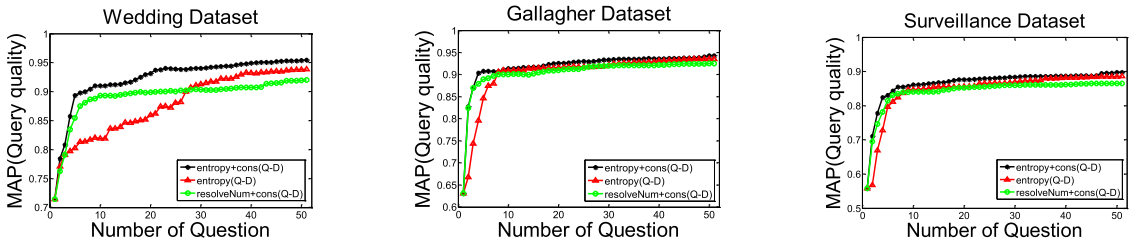
$$AP = \sum_{k=1}^{m} precision(k)\Delta recall(k) \tag{8}$$

where k is the rank in the sequence of returned face list, m is total result number, $precision(k)$ is the precision at cut-off k in the list, and $\Delta recall(k)$ is the change in recall from items $k-1$ to k. We randomly choose queries and perform the process 100 times to compute mean AP as the criteria.

1) Comparison of Question Generation Strategy: Experiments are performed to evaluate the question generation strategy discussed in section 4.2. As illustrated in Figure 8, we compare our approach query-driven entropy with constraints (*entropy+cons(Q-D)*) with other methods. *Entropy* refers to the traditional criteria suggested in paper [7] and [29] which selects samples with the highest uncertainty measured by entropy. *Random* is a naive approach which chooses sample randomly. *Prob* refers to the strategy selecting samples with the highest relevance probability to the target node. Figure 8 illustrates the comparison between the above strategies on the four data collections. It demonstrates the tendency of mean average precision (MAP) against the increase of question number. From the plots, we can see that our approach allow quick improvement to the query answer quality (with just a few tagged face clusters), compared with other strategies. This is because our strategy targets to answer queries rather than tagging the whole dataset.

We also plot the comparison between three query-driven strategies in Figure 9. The method *entropy(Q-D)* computes query-driven entropy without considering constraints. Paper [36] proposed to use the number of resolved faces considering constraints to measure the impact of a cluster node. Here we leverage it in a query-driven manner, referred as *resolvenum + cons(Q-D)*. Analyzing the overall results, we discover that the "*entropy+cons(Q-D)*" approach can achieve the best performance because it considers contextual constraints to compute query-related entropy, which can appropriately estimate the impact of each sample. In contrast, the method "*entropy(Q-D)*" does not illustrate very good performance at the first several iterations due to the lack of consideration of constraints. The approach "*resolvenum + cons(Q-D)*" experiences a significant improvement in the first several iterations, but with slower process after consuming the benefit of constraints.

2) Query-Driven VS. Query-Agnostic Approaches: We also perform experiments to compare query-driven and query-agnostic (or generic) approaches, from the perspective of improving query answer quality and face recognition performance (the goal of traditional face-tagging) respectively. The query-agnostic version of our model is equivalent to (our re-implementation of) the GP-based active-learning approach of Kapoor *et al.* [7], which is a state-of-the-art baseline for interactive face labeling. Figure 10 demonstrates their comparison from the perspective of improving query answer quality. The results illustrate that query-driven approaches



(a) (b) (c)

Fig. 9. Comparison of different Query-driven strategies.

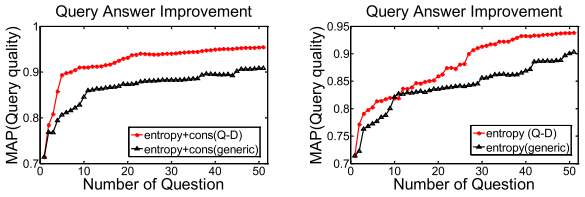


Fig. 10. Query-driven VS. Query-agnostic strategies.

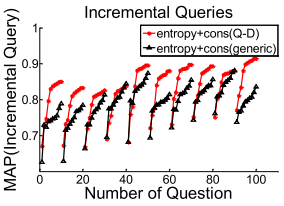


Fig. 11. Performance comparison with incremental queries.

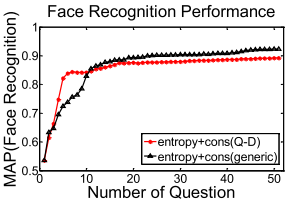


Fig. 12. Comparison of face recognition performance.

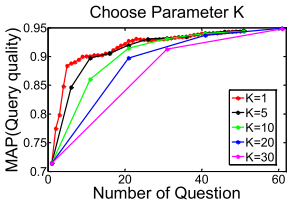


Fig. 13. Parameter Selection.

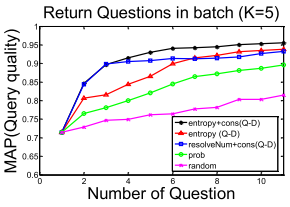


Fig. 14. Run with $K = 5$.

have significant superiority compared to query-agnostic approaches towards quality answer improvement. The advantage of query-driven setting is that we process data as needed, and avoid to waste time to process data that will never be used. Besides, it also aims to choose questions which can maximally help answering queries.

In the real applications, we assume that many different users can present different queries, where the previous labeling results can be accumulated to be leveraged for the following queries. To simulate this behavior, we perform incremental query experiments illustrated in Figure 11. For each query, we assume that users will be satisfied with answers after 9 questions (experiments show that query answer quality can be significantly improved after a few questions). Then we simulate another user will continue to present different queries on the partial-cleaned dataset. We randomly generate 100 query orders to simulate this behavior, and average the performance. The results demonstrate that query-driven approach can achieve much better results with the incremental queries.

We also plot the comparison from the view of face recognition performance in Figure 12. Since the query-driven approaches are designed to favor the improvement of query quality, it will sacrifice the face recognition performance in the entire dataset, to some extent. However, the experiments show that the query-driven approaches still have relatively good performance.

3) Select K Questions in Each Iteration: It is inefficient to ask users only one question in each iteration, therefore, we propose to return K questions in each round. An interesting question is how to choose the size K in each iteration. To choose the appropriate value for parameter K, we select different values (from 1 to 30) and plot the tendency of MAP change. As illustrated in Figure 13, we discover that the larger values of K lead to the slower improvement of query answer quality. This is because if we choose a larger value for K, the classifier cannot be updated promptly, and some unnecessary questions will be asked. However, a small K value will result in too many iterations and computation load. Therefore, to trade off between the two factors, we can set $K = 5$ in the application. Figure 14 illustrates the performance comparison with $K = 5$.

4) Computational Complexity: The bottleneck of our approach is Gaussian process (GP) prediction and factor graph inference. GPs require N^3 operations to invert a N^2 matrix (where N = number of annotated faces), but this inversion can be computed offline and reused during interactive tagging. Our factor graph is sparse. In practice a fixed number of belief

propagation iterations, each $O(N)$ where N = number of unlabeled faces, sufficed. Most importantly, our probabilistic database manager can make use of standard database techniques (blocking, pruning, indexing) to make practical run-times sublinear. For example, blocking on timestamps can significantly reduce computational cost for certain queries (e.g., "find pictures of John from the last week"). In terms of practical run-times, our system takes roughly 0.5 sec (on a commodity desktop) to process an interactive user tag and generate a new question, with 10000 unlabeled faces.

VI. Conclusion

In this paper, we have introduced a query-drive paradigm for face clustering/tagging which can be seamlessly integrated into image analysis/retrieval process, to address the challenges of big data. Our goal is to explore query-driven active learning strategies to achieve accurate query answers with minimum user participation. In our framework, queries have been represented by factor graph to facilitate probabilistic reasoning, where Gaussian process models have been used for probability inference. We have proposed the criteria considering query-driven entropy and contextual constraints to select tagged samples to maximally improve query answer quality. Experiments on real-world datasets have demonstrated the superiority of the proposed query-driven approaches compared to query-agnostic methods towards query answer improvement. The basic idea of this paper is to exploit the semantics of analysis (query) to prune unnecessary vision processing. It can be applied to any application requiring interactive real-time monitoring and analysis of images/visual data - the details and mechanism might differ, but the fundamental concept is still applicable. Our future work is to explore such a generalization.

References

[1] *Cisco Visual Networking Index: Forecast and Methodology, 2013–2018*, Cisco Syst., Inc., San Jose, CA, USA, 2014.

[2] L. Gomes, "Machine-learning maestro michael jordan on the delusions of big data and other huge engineering efforts," *IEEE Spectr.*, Oct. 20, 2014.

[3] Y. Tian, W. Liu, R. Xiao, F. Wen, and X. Tang, "A face annotation framework with partial clustering and interactive labeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2007, pp. 1–8.

[4] M. Wang, B. Ni, X.-S. Hua, and T. Chua, "Assistive tagging: A survey of multimedia tagging with human-computer joint exploration," *ACM Comput. Surv.*, vol. 44, no. 4, 2012, Art. no. 25.

[5] J. Cui, F. Wen, R. Xiao, Y. Tian, and X. Tang, "EasyAlbum: An interactive photo annotation system based on face clustering and re-ranking," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2007, pp. 367–376.

[6] D. Anguelov, K. Lee, S. B. Göktürk, and B. Sumengen, "Contextual identity recognition in personal photo albums," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Minneapolis, MN, USA, Jun. 2007, pp. 1–7. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2007.383057

[7] A. Kapoor, G. Hua, A. Akbarzadeh, and S. Baker, "Which faces to tag: Adding prior constraints into active learning," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep./Oct. 2009, pp. 1058–1065.

[8] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell, "Active learning with Gaussian processes for object categorization," in *Proc. IEEE 11th Int. Conf. Comput. Vis. (ICCV)*, Oct. 2007, pp. 1–8.

[9] H. Altwaijry, D. V. Kalashnikov, and S. Mehrotra, "Query-driven approach to entity resolution," *Proc. VLDB Endowment*, vol. 6, no. 14, pp. 1846–1857, 2013.

[10] R. Ramakrishnan, J. Gehrke, and J. Gehrke, *Database Management Systems*, vol. 3. New York, NY, USA: McGraw-Hill, 2003.

[11] M. Kafai, L. An, and B. Bhanu, "Reference face graph for face recognition," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 12, pp. 2132–2143, Dec. 2014.

[12] L. An, M. Kafai, and B. Bhanu, "Dynamic Bayesian network for unconstrained face recognition in surveillance camera networks," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 3, no. 2, pp. 155–164, Jun. 2013.

[13] L. Zhang, D. V. Kalashnikov, and S. Mehrotra, "A unified framework for context assisted face clustering," in *Proc. ACM Int. Conf. Multimedia Retr. (ICMR)*, Dallas, TX, USA, Apr. 2013, pp. 9–16.

[14] J. Tang, Z. Li, M. Wang, and R. Zhao, "Neighborhood discriminant hashing for large-scale image retrieval," *IEEE Trans. Image Process.*, vol. 24, no. 9, pp. 2827–2840, Sep. 2015.

[15] Z. Li, J. Liu, J. Tang, and H. Lu, "Robust structured subspace learning for data representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 10, pp. 2085–2098, Oct. 2015.

[16] S.-J. Huang, R. Jin, and Z.-H. Zhou, "Active learning by querying informative and representative examples," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 1936–1949, Oct. 2014.

[17] S. Huang, S. Chen, and Z. Zhou, "Multi-label active learning: Query type matters," in *Proc. 24th Int. Joint Conf. Artif. Intell. (IJCAI)*, Buenos Aires, Argentina, Jul. 2015, pp. 946–952. [Online]. Available: http://ijcai.org/Abstract/15/138

[18] J. Tang, Z.-J. Zha, D. Tao, and T.-S. Chua, "Semantic-gap-oriented active learning for multilabel image annotation," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2354–2360, Apr. 2012.

[19] Y.-G. Jiang, J. Wang, X. Xue, and S.-F. Chang, "Query-adaptive image search with hash codes," *IEEE Trans. Multimedia*, vol. 15, no. 2, pp. 442–453, Feb. 2013.

[20] B. Siddiquie and A. Gupta, "Beyond active noun tagging: Modeling contextual interactions for multi-class active learning," in *Proc. 23rd IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, San Francisco, CA, USA, Jun. 2010, pp. 2979–2986.

[21] P. Bosc and O. Pivert, "About projection-selection-join queries addressed to possibilistic relational databases," *IEEE Trans. Fuzzy Syst.*, vol. 13, no. 1, pp. 124–139, Feb. 2005.

[22] S. Choenni, H. E. Blok, and E. Leertouwer, "Handling uncertainty and ignorance in databases: A rule to combine dependent data," in *Proc. 11th Int. Conf. Database Syst. Adv. Appl. (DASFAA)*, Singapore, Apr. 2006, pp. 310–324.

[23] P. Sen, A. Deshpande, and L. Getoor, "PrDB: Managing and exploiting rich correlations in probabilistic databases," *VLDB J.*, vol. 18, no. 5, pp. 1065–1090, 2009.

[24] N. N. Dalvi and D. Suciu, "Efficient query evaluation on probabilistic databases," *VLDB J.*, vol. 16, no. 4, pp. 523–544, 2007.

[25] E. Yilmaz and J. A. Aslam, "Estimating average precision with incomplete and imperfect judgments," in *Proc. ACM CIKM Int. Conf. Inf. Knowl. Manage.*, Arlington, VA, USA, Nov. 2006, pp. 102–111.

[26] M. Seeger, "Gaussian processes for machine learning," *Int. J. Neural Syst.*, vol. 14, no. 2, pp. 69–106, 2004.

[27] C. E. Rasmussen and H. Nickisch, "Gaussian processes for machine learning (GPML) toolbox," *J. Mach. Learn. Res.*, vol. 11, pp. 3011–3015, Mar. 2010.

[28] A. Freytag, E. Rodner, P. Bodesheim, and J. Denzler, "Labeling examples that matter: Relevance-based active learning with Gaussian processes," in *Proc. 35th German Conf. Pattern Recognit. (GCPR)*, Saarbrücken, Germany, Sep. 2013, pp. 282–291.

[29] A. Holub, P. Perona, and M. C. Burl, "Entropy-based active learning for object recognition," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.

[30] *PubFig: Public Figures Face Database*, accessed on Oct. 3, 2015. [Online]. Available: http://www.cs.columbia.edu/CAVE/databases/pubfig/

[31] A. C. Gallagher and T. Chen, "Clothing cosegmentation for recognizing people," in *Proc. IEEE CVPR*, Jun. 2008, pp. 1–8.

[32] *Google Picasa*, accessed on Mar. 15, 2015. [Online]. Available: http://picasaweb.google.com

[33] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.

[34] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Describable visual attributes for face verification and image search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 10, pp. 1962–1977, Oct. 2011.

[35] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[36] L. Zhang, D. V. Kalashnikov, and S. Mehrotra, "Context-assisted face clustering framework with human-in-the-loop," *Int. J. Multimedia Inf. Retr.*, vol. 3, no. 2, pp. 69–88, Jun. 2014.

Liyan Zhang received the Ph.D. degree in computer science from the University of California, Irvine, in 2014. She is currently an Associate Professor with the School of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics. Her research interests include multimedia analysis, computer vision, and database management. She has received the ICMR best paper award in 2013.

Xikui Wang received the B.Sc. degree in computer software engineering from Harbin Engineering University in 2011, and the M.Sc. degree in computer science from Zhejiang University in 2014. He is currently pursuing the Ph.D. degree with the Department of Computer Science, University of California at Irvine. His research interests include database management, data mining and analysis.

Dmitri V. Kalashnikov received the Diploma (*summa cum laude*) degree in applied mathematics and computer science from Moscow State University, Russia, in 1999, and the Ph.D. degree in computer science from Purdue University in 2003. He was an Associate Adjunct Professor of Computer Science with the University of California, Irvine. He is currently a Senior Scientist with AT&T Laboratories Research. In the past, he has also contributed to the areas of spatial, moving-object, and probabilistic databases. His general research interests include databases and data mining. He is currently with a specialization in the areas of entity resolution and data quality and real-time situational awareness. He has received several scholarships, awards, and honors, including an Intel Fellowship and Intel Scholarship. His work is supported by the NSF, DH&S, and DARPA.

Sharad Mehrotra is a Professor with the School of Information and Computer Science, University of California at Irvine, where he is Founding Director of the Center for Emergency Response Technologies. His research interests include various aspects of data management, multimedia, and distributed systems. His recent research focuses on data quality, data privacy, and sensor driven situational awareness systems. He is the recipient of the SIGMOD test of time award in 2012, the DASFAA test of time award in 2013, and numerous best paper awards, including the SIGMOD best paper award in 2004 and the ICMR best paper award in 2013.

Deva Ramanan was an Associate Professor with the University of California, Irvine, prior to joining Carnegie Mellon University (CMU). He is an Associate Professor with The Robotics Institute, CMU. His research interests span computer vision and machine learning, with a focus on visual recognition. He received the David Marr Prize in 2009, the PASCAL VOC Lifetime Achievement Prize in 2010, the NSF Career Award in 2010, the UCI Chancellor's Award for Excellence in Undergraduate Research in 2011, the PAMI Young Researcher Award in 2012, and was selected as one of Popular Science's Brilliant ten researchers in 2012. His work is supported by NSF, ONR, DARPA, and industrial collaborations with the Intel, Google, and Microsoft.

He is on the Editorial Board of the *International Journal of Computer Vision* and an Associate Editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (PAMI). He serves as a Senior Program Committee Member of the IEEE Conference of Computer Vision and Pattern Recognition, the International Conference on Computer Vision, and the European Conference on Computer Vision. He also serves on NSF panels for computer vision and machine learning.