Near Duplicate Identification With Spatially Aligned Pyramid Matching

Dong Xu, *Member, IEEE,* Tat Jen Cham, Shuicheng Yan, *Senior Member, IEEE,* Lixin Duan, and Shih-Fu Chang, *Fellow, IEEE*

***Abstract*—A new framework, termed spatially aligned pyramid matching, is proposed for near duplicate image identification. The proposed method robustly handles spatial shifts as well as scale changes, and is extensible for video data. Images are divided into both overlapped and non-overlapped blocks over multiple levels. In the first matching stage, pairwise distances between blocks from the examined image pair are computed using earth mover's distance (EMD) or the visual word with χ^2 distance based method with scale-invariant feature transform (SIFT) features. In the second stage, multiple alignment hypotheses that consider piecewise spatial shifts and scale variation are postulated and resolved using integer-flow EMD. Moreover, to compute the distances between two videos, we conduct the third step matching (i.e., temporal matching) after spatial matching. Two application scenarios are addressed—near duplicate retrieval (NDR) and near duplicate detection (NDD). For retrieval ranking, a pyramid-based scheme is constructed to fuse matching results from different partition levels. For NDD, we also propose a dual-sample approach by using the multilevel distances as features and support vector machine for binary classification. The proposed methods are shown to clearly outperform existing methods through extensive testing on the Columbia Near Duplicate Image Database and two new datasets. In addition, we also discuss in depth our framework in terms of the extension for video NDR and NDD, the sensitivity to parameters, the utilization of multiscale dense SIFT descriptors, and the test of scalability in image NDD.**

***Index Terms*—Near duplicate detection, near duplicate retrieval, spatially aligned pyramid matching.**

I. Introduction

NEAR duplicate images or videos refer to a pair of images or videos in which one is close to the exact duplicate of the other. There are two different conventional definitions of the term "near duplicate images" [3]: 1) two near duplicate images are perceptually identical, in which the differences comprise of noise, editing operations, small photometric distortions etc, and 2) in a more general case, two near duplicate images are captured from the same 3-D scene, but they are from different viewpoints and may have different illumination conditions. In this paper, we focus on the second case, which is more challenging due to the presence of significant piecewise spatial shifts, scale and photometric variations (see Figs. 1 and 2). Similarly as in [3], we also define two videos as near duplicate videos if they share a large percentage of near-duplicate frames.

There are two related tasks in near duplicate identification (NDI): near duplicate retrieval (NDR) and near duplicate detection (NDD) [29], [31]. NDR aims to find all images or videos that are near duplicates to an input query image or video, which can be formulated as a *ranking* problem. NDD aims to detect all duplicate image or video pairs from all possible pairs from the image or video source, which can be considered as a two-class *classification* problem. NDR has broad applications in copyright infringement detection and query-by-example application, and NDD has been used to link news stories and group them into threads [29] as well as filter out the redundant near duplicate images or videos in the top results from text keywords based web search [24]. As shown in [29] and [31], NDD is more difficult than NDR. Given 600 images with 150 near duplicate pairs, NDR requires only that a duplicity ranked list be generated for each query image based on comparison to the other 599 images. Conversely, NDD involves correctly classifying 150 image pairs from 179 700 candidate pairs, a task which is less forgiving.

Zhang and Chang [29] formulated a stochastic attributed relational graph matching framework for NDI, in which each vertex is used to represent compositional parts from the detected interest points and each edge is used to characterize part relations. The graph parameters were computed through a learning algorithm based on expectation-maximization. However, the graph matching method involves a complex process of stochastic belief propagation and thus identification speed is slow [31]. In [21], Vaiapury *et al.* extended the image matching method in [14] for near duplicate video identification. Based on PCA-scale-invariant feature transform (SIFT), Ke *et al.* [8] developed a point set matching method, while Zhao *et al.* [31] and Wu *et al.* [24] proposed one-to-one symmetric matching algorithms. However, because of the large number of interest points in images (possibly exceeding 1000), direct matching based on interest points is extremely time-consuming and inappropriate for online NDI. While both works also proposed new index techniques to accelerate the matching speed, the

Manuscript received October 13, 2009; revised December 12, 2009. Date of publication May 20, 2010; date of current version August 4, 2010. This work was funded by Singapore A*STAR SERC Grant (082 101 0018). This paper was recommended by Associate Editor E. Izquierdo.

D. Xu, T. J. Cham, and L. Duan are with the School of Computer Engineering, Nanyang Technological University, 639798, Singapore (e-mail: dongxu@ntu.edu.sg; ASTJCham@ntu.edu.sg; S080003@e.ntu.edu.sg).

S. Yan is with the Department of Electrical and Computer Engineering, National University of Singapore, 117576, Singapore (e-mail: eleyans@nus.edu.sg).

S.-F. Chang is with the Department of Electrical Engineering, Columbia University, New York, NY 10027 USA (e-mail: sfchang@ee.columbia.edu).

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/TCSVT.2010.2051286

retrieval performances generally drop after the use of indexing. For NDI, Tang and Gao [19] proposed to use the cascade structure to efficiently integrate different types of features, and Wu *et al.* [25] selected the best feature to cope with different types of image transformations. Recently, Chum *et al.* [3] used a bag-of-words model [12], [18] to deal with SIFT features for large-scale NDI, and proposed a new indexing technique based on *min-Hash* to accelerate the image matching. However, the indexing technique generally degrades the retrieval performance. More importantly, all the existing methods [3], [8], [19], [21], [24], [25], [29], [31] did not explicitly cope with the challenges in NDI from significant spatial shifts and scale variations.

Distances between images or videos are crucial in NDI. Image matching methods based on local SIFT features have demonstrated remarkable performance for object recognition and localization as well as scene classification [5], [10], [12], [13], [18], [30], in which the SIFT descriptors are extracted from densely sampled regions or the salient regions detected by salient region detection algorithms. Several methods, [e.g., earth mover's distance (EMD) [17] and pyramid match kernel (PMK) [5]], were proposed to directly calculate the distance between two images, which may have unequal length of SIFT descriptors. Other methods [18] represented each image as a bag of orderless visual words by quantizing each SIFT descriptor to a visual word with clustering methods (e.g., K-Means). Then χ^2 distance or other distance between two images is calculated based on the *tf* (term frequency) features or *tf-idf* (term frequency-inverse document frequency) features. Zhang *et al.* [30] experimentally reported that the performances of two classification methods based on EMD and χ^2 distances are generally comparable for object recognition.

Recently, multilevel matching methods were also proposed for efficient distance computation and demonstrated promising results in different tasks, such as object recognition, scene classification and event recognition in news video and consumer video [4], [5], [12], [13], [26], [27]. They involved pyramidal binning in different domains (such as feature, spatial and temporal domain) and led to improved performances resulting from information fusion at multiple levels [2], [5], [12], [13], [26]. The first work PMK is much faster than other existing image matching methods (e.g., EMD), because the computational complexity of PMK is linear in the number of features. The prior work spatial pyramid matching (SPM) [12] quantized SIFT descriptors into visual words and employed the fixed block-to-block matching for scene classification based on the observation that the images from the same scene have similar spatial configurations. Observing that one video clip is usually comprised of multiple stages of event evolution, we proposed a multilevel temporal matching technique, referred to as temporal pyramid matching (TPM) [26] here, to recognize events in broadcast news videos. In TPM, one video clip is divided into non-overlapped subclips, and the subclips across different temporal locations may be matched. However, even when TPM is converted to the spatial domain, TPM cannot cope with the full range of spatial shifts because of its strict non-overlapped partitioning scheme. Moreover, TPM does not consider scale variations.

To solve the problems mentioned above, in Section II we propose a two-stage spatially aligned pyramid matching (SAPM) framework. This paper was initially published in [28]. We divide the images into increasingly finer non-overlapped and overlapped blocks at multiple levels, as shown in Fig. 1(a) and (b). Matching is carried out in two stages. In the first stage matching, we compute the pairwise distances between any two blocks from two images with SIFT features, in which EMD based algorithm and the visual word with χ^2 distance based method [30] are employed (We refer to our method as SAPM and SAPM-tf, respectively). The second stage is a block-alignment stage where different block correspondences at the same level as well as across different levels are hypothesized. The output of SAPM or SAPM-tf is a set of 45 characteristic multilevel distances, each of which approximately measures the validity of a specific hypothesis, involving spatial shift and scale change. These distances can be used in a single ranking measure for retrieving near duplicate images. For the NDD task, we also propose a dual sample approach by using the multilevel distances as 45D features and support vector machine (SVM) [1], [22] for binary classification. Additionally, our method can be readily extended to deal with videos by adding a temporal matching stage.

We conducted exhaustive experiments, reported in Section III, to demonstrate the effectiveness of SAPM. These experiments demonstrate that: 1) for image NDR, at each independent level, better performance is achieved by dividing query images into non-overlapped blocks and database images into overlapped blocks, while the best results are obtained by fusing information from multiple levels; 2) for image NDR and NDD, SAPM generally outperforms SPM and TPM; Moreover, our best results are also significantly better than the recent work [31] for Image NDR and NDD; and 3) for video NDR and NDD, SAPM also outperforms SPM and TPM.

The main contributions of this paper include the following.

1) SAPM and SAPM-tf use a novel multilevel matching framework to explicitly address piecewise spatial shifts and scale variations. We develop a new method for video NDI by incorporating a temporal matching stage.
2) We also propose a dual sample approach by using the multilevel distances as features and SVM for binary classification.

II. Spatially Aligned Pyramid Matching

To solve the problem of near duplicate identification, a framework comprising two stages of matching is developed. An image x is divided into 4^l *non-overlapped* blocks at level-l, $l = 0, \ldots, L-1$ in a manner similar to SPM [12], with the block size set as $1/2^l$ of the original image dimensions. As Lazebnik *et al.* [12] noted that performance does not increase beyond three levels, we likewise fix $L = 3$. A finer partition is also used in which overlapped blocks with size equaling $1/2^l$ of the original image dimensions are sampled at a fixed interval, typically $1/8$ of the image width and height. The denser tiling is intended for subimage matching at finer spatial displacements than that of the non-overlapped partition described above. Two kinds of partitions are illustrated in

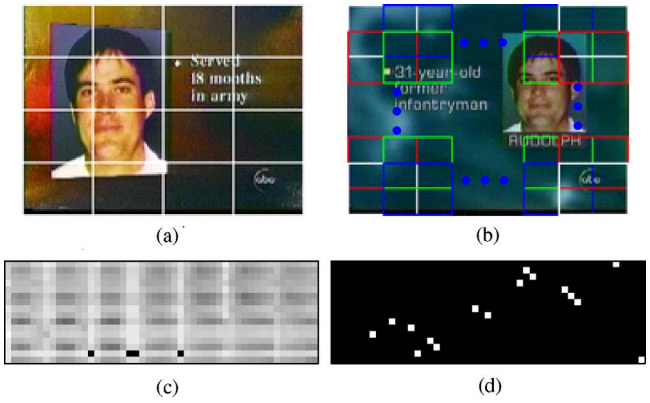


Fig. 1. Illustration of spatially aligned pyramid matching at level-2. (a) and (b) Pair of near duplicate images, which are divided into 4×4 non-overlapped blocks and 7×7 overlapped blocks (as shown with different colors), respectively. (c) 16×49 distance matrix between any two blocks. (d) 16×49 integer flow matrix, which indicates the matching relationship between any two blocks. In this paper, we experimentally compare different overlapped and non-overlapped block partition schemes, and then we observe that the best results for Image NDR are obtained by dividing the query image and the database image into non-overlapped blocks and overlapped blocks, respectively.

Fig. 1(a) and (b). There are a total of five block partition categories, for which we use $p = \{0, 1, 2, 3, 4\}$ to indicate partitions designated as level-0 non-overlapped (L0-N), level-1 non-overlapped (L1-N), level-1 overlapped (L1-O), level-2 non-overlapped (L2-N), and level-2 overlapped (L2-O). The total number of blocks in these five categories are 1, 4, 25, 16, and 49, respectively. We represent image x in the pth partition category as $\{x_r^p,\ r = 1, \ldots, R^p\}$, where x_r^p denotes the rth block and R^p is the total number of blocks. Image y in the qth partition category is represented as $\{y_c^q, c = 1, \ldots, C^q\}$, where y_c^q and C^q are similarly defined. For simplicity, we omit the superscripts p and q unless needed.

A. First Stage Matching

In the first matching stage, the goal is to compute the pairwise distances between any two blocks x_r and y_c. Each block is represented as a bag of orderless SIFT descriptors, and a distance measure is specified that can handle two sets of descriptors with unequal cardinalities. An EMD-based algorithm or χ^2 distance based method [30] is used because of demonstrated effectiveness in several different applications [17], [30].

1) *EMD:* The EMD is used to measure the similarity between two signatures B_1 and B_2. In a manner similar to that described by Zhang *et al.* [30], the set of descriptors in block x_r is clustered to form a signature $B_1 = \{(\mu_1, w_{\mu_1}), \ldots, (\mu_m, w_{\mu_m})\}$, where m is the total number of clusters, μ_i is the center of the ith cluster and w_{μ_i} is the relative size of the ith cluster. EMD is relatively robust to the number of clusters in object recognition, as both demonstrated in our experiments (see Section III-A-1) and in [30]. In this paper, we have three different levels in which m is set as 40, 20 and 20, respectively. The weight w_{μ_i} is equivalent to the total supply of suppliers or the total demand of consumers in the original EMD formulation. The set of descriptors in block y_c is also clustered to form its signature $B_2 = \{(\nu_1, w_{\nu_1}), \ldots, (\nu_n, w_{\nu_n})\}$, where n is the total number of clusters, and ν_i and w_{ν_i} are defined similarly. The ground distance between μ_i and ν_j is defined as d_{ij}, with the Euclidean distance being used in this paper due to simplicity and demonstrated success in [30]. The EMD between x_r and y_c can be computed by

$$D_{rc} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} \widehat{f}_{ij} d_{ij}}{\sum_{i=1}^{m}\sum_{j=1}^{n} \widehat{f}_{ij}} \tag{1}$$

where $\widehat{f}_{ij}$ is the optimal flow that is determined by solving the following linear programming problem:

$$\widehat{f}_{ij} = \arg\min_{f_{ij}} \sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} d_{ij}$$

$$\text{s.t.} \quad \sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} = \min\left(\sum_{i=1}^{m} w_{\mu_i}, \sum_{j=1}^{n} w_{\nu_j}\right) \qquad f_{ij} \geq 0$$

$$\sum_{j=1}^{n} f_{ij} \leq w_{\mu_i} \quad 1 \leq i \leq m \qquad \sum_{i=1}^{m} f_{ij} \leq w_{\nu_j} \quad 1 \leq j \leq n. \tag{2}$$

Given that the Euclidean distance is a metric and the total weight of each block is constrained to be 1, it follows therefore that the EMD distance defined above is a metric [17] (whereby the properties of non-negativity, symmetry and triangle inequality hold). The complexity of EMD is $O(m^3 log(m))$ [17] when the total number of clusters in two blocks are the same, i.e., $m = n$. The distances between all pairs of blocks are obtained after the first stage of matching. Fig. 1 (c) presents a visual representation of the 16×49 distance matrix, where brighter intensities indicate higher distance values between corresponding blocks.

2) χ^2 *Distance:* An alternative method is to construct a global texton vocabulary by clustering the SIFT descriptors from the training set, in which SIFT descriptors are vector quantized into *visual words* with token frequency (tf) features extracted from each block [18]. Suppose the tf features for any two blocks x_r and y_c are represented as $\hat{B}_1 = [\hat{\mu}(1), \hat{\mu}(2), \ldots, \hat{\mu}(H)]^T$ and $\hat{B}_2 = [\hat{\nu}(1), \hat{\nu}(2), \ldots, \hat{\nu}(H)]^T$, where H is the size of the texton vocabulary. The distance D_{rc} between any two blocks x_r and y_c is calculated based on the χ^2 distance as

$$D_{rc} = \frac{1}{2}\sum_{i=1}^{H} \frac{(\hat{\mu}(i) - \hat{\nu}(i))^2}{\hat{\mu}(i) + \hat{\nu}(i)}. \tag{3}$$

In this paper, our multilevel matching methods are referred to as SAPM and SAPM-tf, respectively, in which the first stage of matching uses the EMD based matching algorithm or the χ^2 distance based method.

B. Second Stage Matching

In the second stage of matching, the goal is to have the blocks from one query image x aligned to corresponding blocks in its near duplicate image y. This differs from the fixed block-to-block matching used in SPM [12] in that one block may be matched to another block at a different position and/or scale level in our proposed SAPM and SAPM-tf framework,

allowing it to cope with piecewise spatial translation and scale variation.

Suppose the total number of blocks in x and y are R and C, and the pair-wise distances between any two blocks are D_{rc}, $r = 1, \ldots, R$ and $c = 1, \ldots, C$. The alignment process involves computing a flow matrix $\widehat{F}_{rc}$ comprising binary elements, which represent unique matches between blocks x_r and y_c. For cases when $R = C$, this can be formulated as an integer programming problem embedded within a linear programming framework as suggested by [26]. The following theorem is utilized.

Theorem 1 ([7]): The linear programming problem

$$\widehat{F}_{rc} = \arg\min_{F_{rc}} \sum_{r=1}^{C}\sum_{c=1}^{C} F_{rc}D_{rc} \text{ s.t.}$$

$$0 \le F_{rc} \le 1 \quad \forall r, c \quad \sum_{c=1}^{C} F_{rc} = 1 \quad \forall r \quad \sum_{r=1}^{C} F_{rc} = 1 \quad \forall c \quad (4)$$

will always have an integer optimum solution when solved with the simplex method.[1]

If $R \neq C$, then assuming that $R < C$ without loss of generality, the EMD formulation for block matching has to be broadened to

$$\widehat{F}_{rc} = \arg\min_{F_{rc}} \sum_{r=1}^{R}\sum_{c=1}^{C} F_{rc}D_{rc} \text{ s.t.}$$

$$0 \le F_{rc} \le 1 \quad \forall r, c \quad \sum_{c=1}^{C} F_{rc} = 1 \quad \forall r \quad \sum_{r=1}^{R} F_{rc} \le 1 \quad \forall c. \quad (5)$$

Nevertheless, the formulation in (4) can be re-established from (5) by 1) adding $C - R$ virtual blocks in image x, and 2) setting $D_{rc} = 0$, for all r satisfying $R < r \le C$. Hence for any solution of (4), a flow matrix for (5) can simply be obtained by removing the elements related to the virtual blocks. An integer solution for (4) with virtual blocks can then be obtained via the simplex method as indicated by Theorem 1, from which the integer solution for (5) may be easily extracted. An outcome of this process is illustrated in Fig. 1(d), indicating the matches of the local image areas in two images (e.g., face, text, etc.). With $\widehat{F}_{rc}$, we denote the *distance measure from x to y* as

$$S(x \to y) = \frac{\sum_{r=1}^{R}\sum_{c=1}^{C} \widehat{F}_{rc}D_{rc}}{\sum_{r=1}^{R}\sum_{c=1}^{C} \widehat{F}_{rc}}. \quad (6)$$

Fig. 2 illustrates the differences between SPM [12], TPM [26], and SAPM at level-2, in which three blocks from each query image [i.e., Fig. 2(a)] and their matched counterparts in the near duplicate images [i.e., Fig. 2(b), (c), and (d)] are highlighted, with color of the outlines denoting correspondence. Spatial shifts and scale variations (which also result in spatial shifts) between the near duplicate images are highly obvious. The fixed block-to-block matching approach of SPM [12] is unable to handle such non-proximal spatial changes. To enable comparison with the TPM method, we converted TPM to the spatial domain to obtain the result in Fig. 2(c), which is equivalent to allowing matching between blocks in different spatial positions across the two compared images. However, the TPM results were still poor as the strict non-overlapped block partitioning scheme does not cope with the full range of spatial changes. When compared with SPM and TPM, results from SAPM were much better, which demonstrates its robustness against spatial shifts and scale variations.

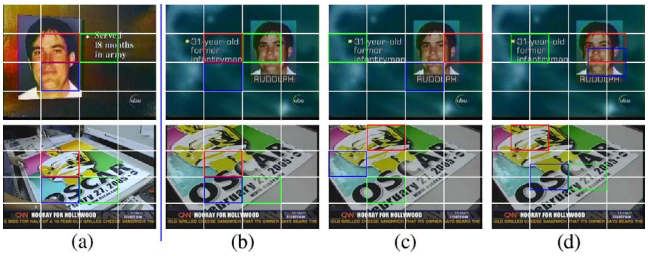


Fig. 2. Comparison of three pyramid matching methods at level-2. Three blocks in the query images [i.e., (a)] and their matched counterparts in near duplicate images [i.e., (b), (c), (d)] are highlighted and associated by the same color outlines.

The following details of our method are noteworthy.

1) SAPM and SAPM-tf preserve some amount of spatial proximity information in the higher levels (i.e., level-1 and level-2). The interest points in one spatial block are restricted to match to only interest points within another block in SAPM and SAPM-tf at a certain level, instead of arbitrary interest points within the entire image as is the case in the classical bag-of-words model (e.g., SPM at level-0) [12], [18].
2) Suppose we divide image x and y into blocks with the pth and qth partition category, respectively, we denote the distance measure from x^p to y^q as $S(x^p \to y^q)$, which can be calculated with (6). There are in total 25 distances[2] addressing different variations between the two images: a) if the query image was divided into non-overlapped blocks (e.g., L2-N) and the corresponding database images were divided into overlapped blocks (e.g. L2-O) at the same level, spatial shifts and some degree of scale change are addressed (e.g., $S(x^{L2-N} \to y^{L2-O})$); b) larger scale variations are considered by matching the query image and the database images at different levels [e.g., $S(x^{L1-N} \to y^{L2-O})$]. Our method can potentially cope with a broad range of scale variations by using denser scale and spatial sampling; Subimage cropping is also considered (e.g., $S(x^{L0-N} \to y^{L1-O})$ and $S(x^{L1-O} \to y^{LO-N})$), which can be treated as a special case of scale variation; c) ideally, SAPM and SAPM-tf can deal with any amount of spatial shift and scale variation by using denser scale and spatial sampling.
3) The following observations can be established from (5): a) if $p = q$, $S(x^p \to y^q) = S(y^p \to x^q)$; b) if $p \neq q$, $S(x^p \to y^q)$ may not be equal to $S(y^p \to x^q)$. This is obvious because x^p includes different blocks from

[1]This problem can be also formulated as a minimum-weight bipartite graph matching problem [15]. However, the main focus of this paper is to propose a general framework for NDI. To be consistent with the EMD in the first stage matching, we formulate it as an integer-flow EMD, which is solved via the simplex method.

[2]There are only 25 distances because $S(x^p \to y^q) = S(y^q \to x^p)$ according to [17] and the above analysis on virtual blocks.

x^q, and also y^p and y^q. The two distances are different because the block partitioning schemes are different, hence we describe the distance measure as *asymmetric*.

4) For purposes of comparison, another possible weighting scheme is also used in which normalizing weights $1/R$ and $1/C$ were applied to the two signatures to replace the unit weights 1 in (5). We denote the *distance measure* from x to y in this case as $\widetilde{S}(x^p \rightarrow y^q)$, which is again asymmetric. We compare the two different weighting schemes for Image NDR in Section III-A-1.

C. Fusion of Information From Different Levels for NDR

Previous work incorporating pyramid matching [5], [12], [13], [26] demonstrated that better results were achieved when multiple resolutions were combined, even in situations when the results obtained using individual resolutions were not accurate. In this paper for NDR, distances from different levels were directly fused

$$S^{\text{Fuse}}(x \rightarrow y) = h_0 S(x^0 \rightarrow y^0) + \sum_{l=1}^{L-1} h_l S(x^{2l-1} \rightarrow y^{2l}) \qquad (7)$$

where h_l is the weight for level-l. As in [26], we considered separately the use of equal weights and unequal weights. Our experiments demonstrated that the results from different weighting schemes are comparable, similar to the findings obtained for TPM [26].

D. Dual Sample Approach for NDD

As a *ranking* problem, NDR can be directly conducted based on the distance measures from SAPM. NDR is thus easier than NDD and amenable to the use of asymmetric distance measures. NDD, conversely, is essentially a two-class classification problem, i.e., an image pair is classified as a duplicate or non-duplicate, which in any case requires symmetric measures. For classification, we need a proper representation for the image pair or video pair. One possibility is to compute the difference vector of features in the two images, but our experiments indicate that such raw differences were insufficient for detecting duplicate images with large variations. Instead, we establish new input features comprising 45 matching distances for the NDD task, with the expectation that near-duplicate image pairs will cluster around the origin in this new feature space while dissimilar image pairs will be far from the origin.

Recall that each weighting scheme in the second stage matching outputs 25 distances, forming a combined 50 distances, except that $S(x^p \rightarrow y^p) = \widetilde{S}(x^p \rightarrow y^p)$ for $p = 0, \ldots, 4$, which means there are only 45 unique distances. In order to bypass the problem of asymmetric distance measures, the kth pair of images (say images x and y) is represented as two samples, denoted as $t_k^1 \in \mathbb{R}^{45}$ and $t_k^2 \in \mathbb{R}^{45}$, where t_k^1 is comprised of the 45 distances from x to y, and t_k^2 is comprised of another 45 distances from y to x. The same class label (1 or 0) is assigned for t_k^1 and t_k^2. Denote T as the total number of image pairs in the training set, the training samples are then represented as $\{t_1, \ldots, t_{2T}\} = \{t_1^1, t_1^2, \ldots, t_T^1, t_T^2\}$. Subsequently, classification is done through SVM. In the testing stage, SVM outputs two decision values η_k^1 and η_k^2 for the kth pair of images, and then the final decision value is computed via a sigmoid-like function

$$\eta_k = \frac{0.5}{1 + \exp(-\eta_k^1)} + \frac{0.5}{1 + \exp(-\eta_k^2)}. \qquad (8)$$

While other approaches to handle the asymmetric matching may be possible (e.g., average or aggregation in a long vector), the dual sample approach mentioned above is preferred so that patterns associated with individual feature of the asymmetric pair can be preserved and used to detect near duplicates.

E. Extension to Video NDI With Temporal Matching

In the case of video NDI, a third stage of temporal matching is added after the previous two spatial matching stages. Following the same approach as in spatial matching, EMD is employed here as well. Suppose one video clip V_1 comprises $\{x(1), x(2), \ldots, x(M)\}$, where $x(i)$ is the ith frame and M is the total number of frames of V_1, while another video clip V_2 comprises $\{y(1), y(2), \ldots, y(N)\}$, where N and $y(j)$ are similarly defined. After the two stages of SAPM matching, pair-wise distances $S(x(i) \rightarrow y(j))$ between every two frames $x(i)$ and $y(j)$ are obtained. The d_{ij} distances in (1) and (2) are set as $S(x(i) \rightarrow y(j))$, and the existing EMD framework can be directly employed to compute the distances from V_1 to V_2. Fig. 3(a) and (e) displays sample frames from two video sequences, while Fig. 3(b) shows the pairwise distance matrix between any two frames obtained from the SAPM framework.

We also consider the two possible weighting schemes of normalized weights $(1/M, 1/N)$ and unit weights for temporal matching. With normalized weights, the flow matrix comprises continuous elements. With unit wights 1, the flow matrix comprises only binary elements 0 or 1 (see Theorem 1). The flow matrices under these two schemes are also shown in Fig. 3(c) and (d). Our experiments demonstrate that the use of unit weights is comparable or slightly better than normalized weights in temporal matching for Video NDR. Since we defined two videos as near duplicate if they shared a large percentage of near-duplicate frames, the temporal matching scheme with unit weights is logically more appropriate. The distances from the two weighting schemes are combined as features in video NDD.

Suppose the complexity of matching two images is $O(H)$ and the total numbers of frames in two video clips are the same (i.e., $M = N$), the complexity of matching two video clips is $O(HM^2 + M^3 \log(M))$, where $O(HM^2)$ is the complexity to calculate the pair-wise distance between any two images and $O(M^3 \log(M))$ is the complexity of temporal matching using EMD. Currently, the most computationally expensive component in our matching framework is the EMD calculation due to its super-cubic complexity. In the future, we plan to use the recently proposed fast earth mover's distances [16] for speedup because it is reported that the distance can be calculated by an order of magnitude faster than the original implementation [17]. Large-scale video NDI may be addressed through a two-step approach in the future: 1) we represent one video sequence as one keyframe and employ SAPM (or SAPM-tf) to rapidly filter out a large portion of irrelevant

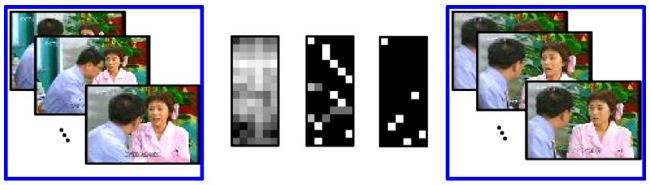


Fig. 3. Illustration of temporal matching in Video NDI. (a) and (e) Two video clips and their frames. (b) Pair-wise distance matrix between any two frames from SAPM based spatial matching. (c) and (d) Two flow matrices under two different weighting schemes in temporal matching.

videos; and 2) re-rank the remaining videos using SAPM (or SAPM-tf) + temporal matching (TM) based on all the frames of video clips to further improve the identification performance.

III. Experiments

Extensive experiments were conducted to test the effectiveness of SAPM and SAPM-tf. The primary dataset used is the Columbia Near Duplicate Image Database [29], in which the images were collected from TRECVID 2003 corpus [20]. We additionally annotated another near duplicate image database, referred to in this paper as the New Image Dataset, in which the images were chosen from key-frames of the TRECVID 2005 and 2006 corpus [20]. Very significantly, the New Image Dataset contains much greater variation in spatial translations and scale variations as compared to the Columbia dataset. In both datasets, there are 150 near duplicate pairs (300 images) and 300 non-duplicate images. The new data set is also more challenging and realistic compared to synthesized data used in [8], [11] as it is based on real broadcast news rather than edits of the same image.

Likewise, we also annotated a near duplicate video database, referred to in this paper as the New Video Dataset. In TRECVID, the temporal boundaries for each shot were provided by NIST, which were used to generate the candidate video clips. Each candidate shot was initially sampled at two frames per second to extract image frames, followed by having a human annotator review all extracted frames to refine the start/end boundaries of the shot. In total, there are 50 video near duplicate pairs (100 video clips) and 200 non-duplicate videos. We will make these newly annotated data sets publicly available.

For performance evaluation, we used all near duplicate image pairs as queries. For each query, other images were ranked based on computed distances. The retrieval performance was evaluated based on the probability of successful top-k retrieval [29], [31], i.e., $P(k) = Q_c/Q$, where Q_c is the number of queries that rank their near duplicates within the top-k positions, and Q is the total number of queries.

As NDD can be treated as a two-class classification problem, performance evaluation can be done through the use of equal error rate (EER) and non-interpolated Average Precision (AP) [20]. EER measures the accuracy at which the number of false positives and that of false negatives are equal. AP has been used as the official performance metric in TRECVID, and corresponds to the multi-point average precision value of a precision-recall curve, and incorporates the effect of recall when AP is computed over the entire classification result set. For SAPM and SAPM-tf, we defaultly extract SIFT features via the Laplacian detector [14], except that we use multiscale dense SIFT descriptor in Section III-A-3. All the experiments are performed on a Pentium IV-3.0GHZ server with 16 GB RAM. The notation "L2-N → L2-O" is used to indicate a match, in which the query and database images are L2-N and L2-O, respectively. We may also omit L2 and use "N → O" to indicate matching at any level.

TABLE I
Top-1 Retrieval Performance (%) With Different Block Partition Categories on the Columbia Database

Query Image	Image in Database				
	L0-N	L1-N	L1-O	L2-N	L2-O
L0-N	**73.7/73.7**	48.0/37.7	65.3/51.7	25.3/6.3	32.7/9.7
L1-N	39.0/61.3	74.7/74.7	**78.0**/71.7	62.0/20.7	65.7/25.3
L1-O	52.7/61.0	56.3/62.3	76.0/76.0	13.0/14.3	54.7/23.3
L2-N	16.7/46.7	46.0/63.3	65.3/65.7	69.7/69.7	**79.0**/65.7
L2-O	17.0/49.3	40.0/66.7	67.0/71.3	52.0/64.3	71.0/71.0

Each table cell reports the performances (with unit weights)/(with normalized weights).

A. Image Near Duplicate Retrieval

The experiments in this section are to investigate how well SAPM performs under different configurations and parameters, and also to compare the effectiveness of SAPM to the existing SPM and TPM methods in the task of Image NDR. Testing was also carred out on SAPM-tf with the SIFT descriptors extracted via Laplacian detector, as well as SAPM-tf-Dense with the multiscale dense SIFT descriptors. We further compared SAPM-tf and SAPM-tf-Dense to the recent work reported in [31].

1) *Comparison of SAPM Under Different Configurations and Parameters for Image NDR:* SAPM was evaluated for the Image NDR task under different overlapped and non-overlapped block partition schemes as well as two weighting schemes in the second stage matching (see Section II-B). Tables I and II show the top-1 retrieval performances from two weighting schemes (unit and normalized weights) on the Columbia database and New Image Dataset, respectively. In both databases, the best results at level-1 and level-2 (shown in bold) are obtained from "L1-N → L1-O" and "L2-N → L2-O" with *unit weights*, respectively; thus these are used as the ***default configuration*** in later experiments.

Four options are available at each level assuming the use of unit weights ("N → N," "O → N," "N → O," and "O → O"): 1) "N → N" restricts shift distances to be integral multiples of block size, and does not cope with shifts that are smaller than the block size; 2) "O → N" may contain some query blocks which are matched to empty blocks in the integer-flow EMD solution, resulting in information loss from the unmatched query blocks; 3) "N → O" is the most natural matching scheme, which is analogous to the block-based motion estimation method used in the MPEG video compression standard [23], in which a new image frame is di-

vided into non-overlapped blocks and optimal reference blocks are searched over all possible pixel locations in previous frame. In the integer-flow EMD solution, the information content in some blocks in the database image may not be utilized if they are matched to padded empty blocks. This is acceptable since our objective is to find duplicates of the query image, not the database image; and 4) conceptually, "O → O" should provide the most flexible matching, and its performance indeed is the second highest among the above four options (as shown in Tables I and II). However, it is still less effective than "N → O." One possible explanation is that as more noisy matched blocks are included in this method, the normalizing total flow [denominator in (6)] is also increased (e.g., from 16 to 49 at level 2). This results in reduced detection rates for duplicates of partial image matches, due to normalized lower EMD matched scores.

A further observation that can be made is that the matching distances within the same level are consistently better than those across different levels, especially for the Columbia Dataset. However, for the New Image Data Set, utilizing cross-level distances with unit weights results in significant improvement when compared to the performance on the Columbia Dataset. This can be attributed to the fact that the Columbia Dataset has a substantially lower range of scale variation compared to the New Image Dataset. In practice, cross-level distances are better at handling greater scale variation, while within-level distances work better with a smaller range of scale variation. Ideally, SAPM and SAPM-tf can deal with any scale variations with denser scales and grid spacings. Additionally, the results presented in each diagonal cell of Tables I and II are the same, because the distances computed by the two different weighting schemes are expected to be identical in these cases.

Finally, we discuss the performance of SAPM under different parameters. We fixed $L = 3$ based on the empirical observation in [12] that the performance does not increase when extending beyond three levels. As in [12], we also set the total number of non-overlapped blocks as 1, 4 and 16 at three levels, respectively. We also need to determine the total number of clusters m (or n) in each block at three levels and the total number of overlapped blocks $nBlk$ at level-1 and level-2. Here we take the matching "L1-N → L1-O" on Columbia Dataset as an example for discussion. We set $m = n = 10, 20$ and 40 and set $nBlk$ as 3×3 and 5×5. When $nBlk$ is set as 3×3, the overlapped blocks are sampled at a fixed interval $\frac{1}{4}$ of the image width and height. Table III reports the top-1 retrieval performance and the average processing time of SAPM for matching a pair of images. We observe that: 1) SAPM is relatively robust to $nBlk$ and m (or n); and 2) SAPM generally achieves better performance, if we increase the total number of clusters m (or n) from 10 to 40 or increase the total number of overlapped blocks $nBlk$ from 3×3 to 5×5. We observe similar trend at other levels and on New Image Dataset. We therefore set $nBlk$ as 5×5 in this paper. Note SAPM can potentially achieve better performance with more overlapped blocks, but the computational cost is increased as well. Considering the tradeoff of the effectiveness and efficiency, we set $m = 20$ for SAPM at level-1 and level-2. We also set $m = 40$ for SAPM at level-0 because the average processing time with $m = 40$ at level-0 is much less than that of higher levels (see Table VI).

TABLE II
TOP-1 RETRIEVAL PERFORMANCE (%) WITH DIFFERENT BLOCK PARTITION CATEGORIES ON NEW IMAGE DATASET

Query Image	Image in Database				
	L0-N	L1-N	L1-O	L2-N	L2-O
L0-N	**82.0/82.0**	62.3/36.0	77.0/55.3	29.3/5.7	41.3/7.0
L1-N	51.0/72.3	79.3/79.3	**87.7**/80.3	71.7/17.3	79.3/21.3
L1-O	68.0/69.7	65.0/70.0	84.3/84.3	13.0/14.7	70.3/22.3
L2-N	26.3/48.0	53.0/67.3	78.0/76.3	64.7/64.7	**82.7**/68.7
L2-O	28.7/51.7	52.0/71.0	79.3/80.7	46.3/64.3	78.3/78.3

Each table cell reports the performances (with unit weights)/(with normalized weights).

TABLE III
TOP-1 RETRIEVAL PERFORMANCE (%) AND THE AVERAGE PROCESSING TIME (MS) FOR MATCHING A PAIR OF IMAGES OF SAPM (''L1-N → L1-O'') ON COLUMBIA DATASET

	Top-1 Retrieval Performance (%)		Average Processing Time (ms)	
m & n	$nBlk = 3 \times 3$	$nBlk = 5 \times 5$	$nBlk = 3 \times 3$	$nBlk = 5 \times 5$
10	76.3	77.0	2.1	5.9
20	76.3	78.0	8.8	24.4
40	78.3	78.3	43.1	120.9

TABLE IV
TOP-1 RETRIEVAL PERFORMANCE (%) COMPARISON OF SAPM, SPM AND TPM FROM SINGLE LEVEL AND MULTIPLE LEVELS ON COLUMBIA DATABASE

	L0-N → L0-N	L1-N → L1-N (or L1-O)	L2-N→L2-N (or L2-O)
Single-level (SPM)	73.7	76.3	73.3
Single-level (TPM)	73.7	74.7	69.7
Single-level (SAPM)	73.7	**78.0**	**79.0**
Multilevel (SPM)		76.7 / 76.0	78.0 / 77.3 / 77.7
Multilevel (TPM)		75.0 / 75.3	75.7 / 74.7 / 75.3
Multilevel (SAPM)		77.7 / **78.0**	79.3 / 80.0 / **80.7**

In the last three rows, the first number is from the equal weighting scheme and the last one or two numbers in each cell are from the unequal weighting scheme when fusing multiple levels.

2) *Comparison of SAPM With SPM and TPM for Image NDR:* We compared SAPM with SPM and TPM for cases when matching was done at individual levels as well as when fusing multiple levels. We tried two weighting schemes for cases when multiple resolutions are fused: 1) equal weights, $h_0 = h_1 = h_2 = 1$; and 2) unequal weights: $h_0 = 1$ and $h_1 = 2$ for fusing only the first two levels as well as $h_0 = h_1 = 1, h_2 = 2$ and $h_0 = 1, h_1 = h_2 = 2$ for fusing all three levels. Note that above equal weights and unequal weights based methods were similarly suggested in the prior work [5], [12], [26]. While it is possible to learn the optimal weights to fuse the information from multiple levels (see [6]), the motivation of this paper is to propose a general multilevel spatial matching framework for NDI, rather than developing new fusion algorithms.

The results are listed in Tables IV and V, in which the default configuration is used for SAPM at level-1 and 2. The following observations can be made.

TABLE V
TOP-1 RETRIEVAL PERFORMANCE (%) COMPARISON OF SAPM, SPM AND TPM FROM SINGLE LEVEL AND MULTIPLE LEVELS ON NEW IMAGE DATASET

	L0-N → L0-N	L1-N → L1-N (or L1-O)	L2-N→L2-N (or L2-O)
Single-level (SPM)	82.0	82.0	71.0
Single-level (TPM)	82.0	79.3	64.7
Single-level (SAPM)	82.0	**87.7**	**82.7**
Multilevel (SPM)		85.3 / 84.3	84.7 / 81.3 / 82.3
Multilevel (TPM)		84.0 / 82.7	83.0 / 79.3 / 80.7
Multilevel (SAPM)		87.3 / **88.0**	**88.3** / 88.0 / 88.0

In the last three rows, the first number is from the equal weighting scheme and the last one or two numbers are from the unequal weighting scheme when fusing multiple levels.

1) When compared with SPM and TPM, the results from SAPM are better at a single level (i.e., level-1 or level-2), which demonstrates that the second stage matching based on the integer-flow EMD in SAPM (see Section II-B) can effectively cope with the spatial shift.
2) For SAPM, in most cases better performance can be achieved when multiple resolutions are combined, even for resolutions that are independently poor; moreover, there is no single level that is universally optimal in the two databases. Therefore, the best solution is to combine the information from multiple levels in a principled way, as reported in the prior work [5], [12], [26]. SAPM also outperforms SPM and TPM after multilevel fusion.
3) For SAPM, the results from different weighting schemes are generally comparable, similar to the findings from [26].
4) The results from TPM are worse than SPM, a possible explanation of which is that near duplicate images retain somewhat similar spatial layouts, which fits the SPM model. We also observed that the best result from fusing the first two levels is better than that from fusing all the three levels for SPM and TPM in New Image Dataset, which is consistent with prior work [12], [26]. In practice, correlation of features at mid to lower spatial frequencies can be used to adequately detect near image duplicates.

Fig. 4 compares the best top-30 retrieval performance from SAPM, SPM, and TPM. Again, we observe that SAPM consistently outperforms SPM and TPM. Fig. 5 shows the unsuccessful cases of SAPM, in which the three rows from the top to the bottom show the query image, the correct near duplicate database image and the wrongly retrieved top-1 database image, respectively. We observe that the near duplicate pairs contain significant spatial shifts and scale variations, making Image NDR a challenging task. Another possible reason for failure is that SIFT features may not be discriminative enough for characterizing the complex scenes and objects in these cases.

Finally, we analyze the algorithmic complexity and the average processing time of SAPM. Suppose the total number of signatures are the same, i.e., $m = n$, then the complexity of EMD is $O(m^3 log(m))$ [17]. In Table VI, we report the average processing time for matching a pair of images of SAPM, SPM,

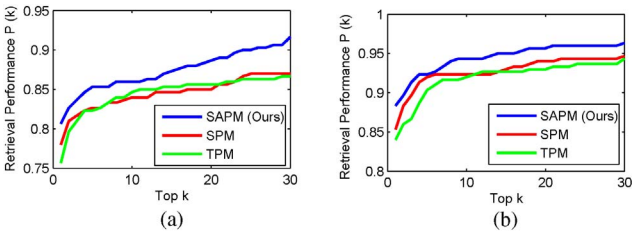


Fig. 4. (a), (b) Comparison of best top-30 retrieval performance from SAPM, SPM, and TPM.

Fig. 5. Unsuccessful cases of SAPM. The three rows (from the top to the bottom) show the query images, the correct near duplicate database images, and the wrongly retrieved top-1 database images, respectively.

and TPM. We observe that SAPM is slower than SPM and TPM at level-1 and level-2, because it is time-consuming to calculate the distances between every pair of blocks of the two images in the first stage of EMD matching. In the following, we show that the alternative method SAPM-tf can significantly accelerate SAPM.

3) *SAPM-tf and SAPM-tf-Dense for Image NDR:* We also test the performance of SAPM-tf, in which χ^2 distance based method is used in the first stage matching (see Section II-A-2). In practice, we employ K-Means on randomly selected 60 000 descriptors from the training images to obtain a global texton vocabulary. Similar to [12], we set the size of the global vocabulary as 200 at level-2 and 400 at level-0 and level-1. Note that we do not set the size of the global vocabulary as 400 at level-2 because we observe that the tf feature is very sparse in this case, which degrades the performance of SAPM-tf. Table VII reports the top-1 retrieval performance and the average processing time for matching a pair of images of SAPM-tf, in which we use the matching "L0-N → L0-N," "L1-N → L1-O," and "L2-N → L2-O" for three individual levels again and we employ the equal weights to fuse the results from all three levels. From Table VII, we can observe that: 1) SAPM-tf achieves comparable or better performance, when compared with SAPM; and 2) SAPM-tf is much faster than SAPM, the total average processing time from three levels is, respectively, reduced from 203.7 to 2.9, and from 198.0 to 2.7 on Columbia Dataset and New Image Dataset.

As a general image matching framework, SAPM and SAPM-tf can also cope with multiscale dense SIFT descriptors. Considering that SAPM-tf is much faster than SAPM, we take SAPM-tf as an example to deal with the dense SIFT descriptors, and refer to this method as *SAPM-tf-Dense*. In

TABLE VI
COMPARISON OF THE AVERAGE PROCESSING TIME (MS) FOR MATCHING A PAIR OF IMAGES OF SAPM, SPM, AND TPM ON COLUMBIA DATASET AND NEW IMAGE DATASET

	Columbia Image Dataset				New Image Dataset			
	Level-0	Level-1	Level-2	Total	Level-0	Level-1	Level-2	Total
SPM	1.2	1.0	3.6	5.8	1.2	1.0	3.5	5.7
TPM	1.2	3.9	57.4	62.5	1.2	3.9	55.1	60.2
SAPM	1.2	24.4	178.1	203.7	1.2	24.6	172.2	198.0

TABLE VII
TOP-1 RETRIEVAL PERFORMANCE (%) AND THE AVERAGE PROCESSING TIME (MS) FOR MATCHING A PAIR OF IMAGES OF SAPM-TF ON COLUMBIA DATASET AND NEW IMAGE DATASET

	Top-1 retrieval performance (%)				Average processing time (ms)			
	Level-0	Level-1	Level-2	Multilevel	Level-0	Level-1	Level-2	Total
Columbia Dataset	73.3	80.0	78.3	**80.7**	0.01	0.4	2.5	2.9
New Image Dataset	84.7	90.0	81.3	**90.3**	0.01	0.4	2.3	2.7

TABLE VIII
TOP-1 RETRIEVAL PERFORMANCE (%) AND THE AVERAGE PROCESSING TIME (MS) FOR MATCHING A PAIR OF IMAGES OF SAPM-TF-DENSE ON COLUMBIA DATASET AND NEW IMAGE DATASET

	Top-1 retrieval performance (%)				Average processing time (ms)			
	Level-0	Level-1	Level-2	Multilevel	Level-0	Level-1	Level-2	Total
Columbia Dataset	76.3	80.7	81.3	**82.3**	0.01	0.3	3.1	3.4
New Image Dataset	88.0	92.7	**94.7**	**94.7**	0.01	0.3	3.1	3.4

this paper, the dense SIFT descriptors are extracted over a grid with spacing of four pixels and at three scales. The total number of SIFT descriptors is 14 400 for an image with 320×240 pixels. Similarly as in [12], we employ K-Means on randomly selected 180 000 descriptors from three scales of the training images to obtain a global texton vocabulary, and we set the size of the global vocabulary as 400 at three levels. Table VIII reports the top-1 retrieval performance of SAPM-tf-Dense on Columbia Dataset and New Image Dataset, in which again we use the default matching "L0-N $\rightarrow$ L0-N," "L1-N $\rightarrow$ L1-O," and "L2-N $\rightarrow$ L2-O" for three individual levels and we employ the equal weights to fuse the results from all three levels. We observe that SAPM-tf-Dense outperforms SAPM-tf, which demonstrates that multiscale dense SIFT descriptors are more effective for Image NDR.

4) *Comparison with [31]:* In Table IX, we compare our best results with the recent work [31]. In representing the average processing time of our methods SAPM-tf and SAPM-tf-Dense, the first value only takes into account the average processing time purely for online image matching. The other processes, including vocabulary construction with K-means clustering, quantization of SIFT descriptors into visual words, and tf feature extraction, can be done off-line. The second value in the parentheses includes the average processing time of these off-line processes. As performed in [31], we extract SIFT descriptors from salient regions detected by DoG interest region detector [14]. In average, each image has thousands of interest points. The one-to-one symmetric matching (OOS) involves complex process of matching individual interest points in two images and we observe that it is computationally prohibitive for OOS[3] to directly match two images with more than ten thousands of interest points (e.g., the densely extracted features). From Table IX, we can observe that: 1) SAPM-tf and SAPM-tf-Dense outperform OOS [31], and 2) SAPM-tf and SAPM-tf-Dense are much faster, when compared with OOS [31].

B. Image Near Duplicate Detection

We compared SAPM with SPM, TPM and the algorithm in [31] for Image NDD. For baseline algorithms SPM, TPM, SAPM, and OOS, we use the three distances computed at three independent levels from SPM, TPM and SAPM (with default configurations) or use the distances from OOS as input features, and further apply SVMs for classification. We also report the results by using 45 distances as the features, which is referred to as SAPM-45D. For SAPM and SAPM-45D, the dual sample approach discussed in Section II-D is used to cope with the asymmetric matching. In addition, we report the results based on the NDD method in [31], which is referred to as OOS-Hist. In OOS-Hist, the histograms, which are constructed by counting the number of matched SIFT descriptors of any pair of images at several particular ranges of orientations, are used as features for the subsequent SVM classification (see [31] for more details).

We randomly partition the data into training and test sets. All experiments are repeated ten times with different randomly selected training and test samples, and the means and standard

[3]While Zhao *et al.* [31] also proposed a hash indexing method for speedup, the retrieval performance generally drops. Moreover, with a Pentium IV-3.0GHZ machine, the reported average processing time of OOS after speedup on the same Columbia Dataset is still much worse than our SAPM-tf.

TABLE IX

COMPARISON OF TOP-1 RETRIEVAL PERFORMANCE (%) AND THE AVERAGE PROCESSING TIME (MS) FOR MATCHING A PAIR OF IMAGES OF OOS [32], SAPM-TF, AND SAPM-TF-DENSE ON THE COLUMBIA DATASET AND THE NEW IMAGE DATASET

	Top-1 Retrieval Performance (%)			Average Processing Time (ms)		
	OOS [31]	SAPM-tf	SAPM-tf-Dense	OOS [31]	SAPM-tf	SAPM-tf-Dense
Columbia Dataset	79.0	80.7	82.3	1011.0	2.9 (4.0)	3.4 (7.2)
New Image Dataset	87.0	90.3	94.7	567.2	2.7 (3.8)	3.4 (7.0)

For the average processing time of SAPM-tf and SAPM-tf-Dense, the first value includes only the processing time purely for online matching, whereas the second value in the parentheses includes the average processing time of the off-line processes. Note that the top-1 retrieval performance of OOS on Columbia Dataset is directly from [32].

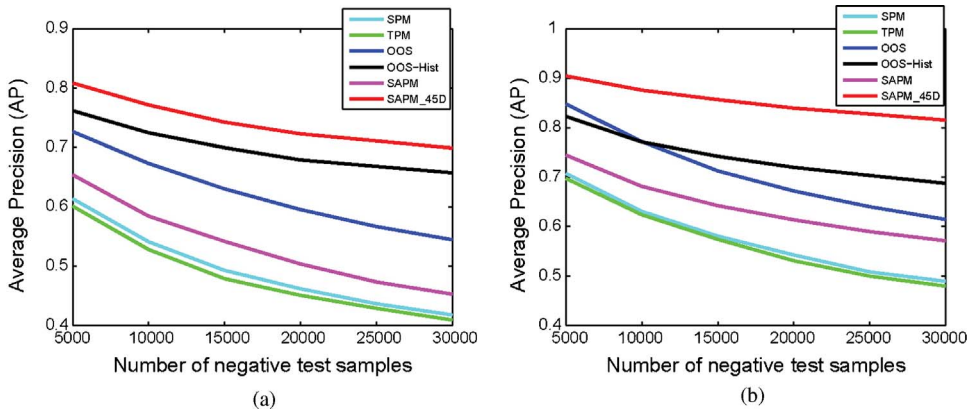


Fig. 6. Average Precision (AP) variations of SAPM-45D, SAPM, SPM, TPM, OOS and OOS-Hist [31] with different number of negative test samples. (a) Columbia Dataset. (b) New Image Dataset.

TABLE X

EER % COMPARISON OF DIFFERENT ALGORITHMS FOR IMAGE NDD ON THE COLUMBIA DATABASE AND NEW IMAGE DATASET

Algorithm	Columbia Database	New Image Dataset
SPM	85.1 ± 2.0	89.7 ± 2.4
TPM	85.0 ± 2.5	90.0 ± 1.7
OOS	87.7 ± 1.8	94.3 ± 1.3
OOS-Hist [31]	88.1 ± 2.0	92.3 ± 1.3
SAPM	87.6 ± 1.9	91.3 ± 1.7
SAPM-45D	**92.1 ± 1.0**	**95.9 ± 1.3**

TABLE XI

AVERAGE PRECISION (AP %) COMPARISON OF DIFFERENT ALGORITHMS FOR IMAGE NDD ON THE COLUMBIA DATABASE AND NEW IMAGE DATASET

Algorithm	Columbia Database	New Image Dataset
SPM	61.4 ± 5.9	70.7 ± 3.8
TPM	60.1 ± 8.2	69.7 ± 4.7
OOS	72.7 ± 6.8	84.8 ± 7.6
OOS-Hist [31]	76.2 ± 5.0	82.3 ± 3.4
SAPM	65.4 ± 4.2	74.5 ± 2.1
SAPM-45D	**80.8 ± 5.4**	**90.4 ± 1.6**

deviations are reported. In each run, we use 60 positive and 240 negative samples for SVM training. The total numbers of positive and negative test samples are 90 and 5000, respectively. We compare SAPM-45D and SAPM with other methods in terms of EER and AP in Table X and XI. From Table X and XI, we observe that: 1) by using only three distances as features, the baseline SAPM is better than SPM and TPM for Image NDD; and 2) the best results are from SAPM-45D, demonstrating that SAPM-45D is a robust Image NDD method.

Finally, we test the scalability of SAPM-45D and other algorithms for Image NDD. We set the total number of negative test samples from 5000 to 30 000 at intervals of 5000. Fig. 6 plots the AP variations of SAPM-45D and other methods with different numbers of negative test samples. From Fig. 6, we observe that: 1) the baseline SAPM is consistently better than SPM and TPM, and SAPM-45D consistently achieves the best results for Image NDD, and 2) the APs of all the methods decrease, when more negative test samples are employed.

C. Video Near Duplicate Retrieval and Detection

For Video NDI, the matching methods SPM, TPM and SAPM are combined with TM, as described in Section II-E, and are denoted as SPM+TM, TPM+TM and SAPM+TM. In SAPM+TM, we use the default configuration when performing SAPM.

For Video NDR, we compared the two weighting schemes using unit and normalized weights in the third stage temporal matching. The earlier mentioned unequal weighting scheme is used in fusing the first two levels. The results are shown in Fig. 7. We observe that SAPM+TM outperforms SPM+TM and TPM+TM at level-1, and the best fused result from the first two levels is also better. Our experiments also demonstrate that the use of unit weights is comparable or slightly better than normalized weights in temporal matching.

In Video NDD, the distances from the first two levels as well as two weighting schemes in temporal matching are used

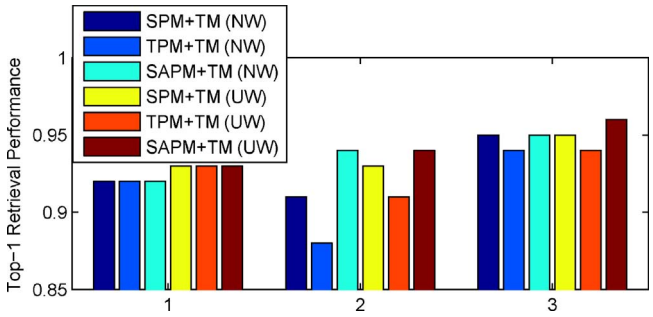


Fig. 7. Comparison of top-1 retrieval performance from SAPM+TM, SPM+TM and TPM+TM with normalized weight (NW) and unit weight (UW) in temporal matching on New Video Dataset. 1: Single-level L0-N → L0-N; 2: Single-level L1-N → L1-N (or L1-O); 3: Multilevel.

TABLE XII
EER % COMPARISON OF DIFFERENT ALGORITHMS FOR VIDEO NDD ON NEW VIDEO DATASET

Algorithm	New Video Dataset
SPM+TM	94.3 ± 2.7
TPM+TM	94.0 ± 2.6
SAPM+TM	**95.7 ± 2.7**

as features. Based on the 4-D feature vector, SVM is used for classification. In SAPM, the dual sample approach discussed in Section II-D is used again to deal with the asymmetric matching. Again, we randomly partition the data into training and test sets, and all experiments were repeated ten times with different randomly selected training and test samples. In each run, we use 20 positive and 80 negative samples for SVM training. The total number of positive and negative test samples are 30 and 5000, respectively. The means and standard deviations are reported in Table XII. From it, we observe that SAPM+TM also outperforms SPM+TM and TPM+TM for Video NDD.

IV. CONCLUSION AND FUTURE WORK

In this paper, a multilevel spatial matching framework with two stage matching has been proposed to deal with spatial shifts and scale variations for image-based near duplicate identification. We further conducted an additional temporal matching stage after spatial matching to effectively compute the distances between two videos. For the task of NDD, we proposed a dual sample approach to cope with the asymmetric matching. The extensive experiments on the Columbia near duplicate database as well as two new datasets clearly demonstrate the proposed multilevel matching framework outperforms the existing methods, exhibiting robustness to different variations. We also conducted in-depth investigation of various aspects of our framework such as parameter selection, the utilization of multiscale dense SIFT descriptors and the test of scalability in image NDD. To the best of our knowledge, this paper and our initial conference version [28] are the first general multilevel spatial matching framework for both image and video NDI.

In the future, we plan to extend our multilevel matching framework to better cope with large-scale video NDI by using other effective features (e.g., space-time features [9]) as well as developing new efficient video matching methods. We will also investigate new algorithms to adaptively divide images (*reps.* videos) into overlapped or non-overlapped blocks (*reps.* space-time volumes) based on the density of SIFT features (*reps.* space-time features [9]).

REFERENCES

[1] C. Burges, "A tutorial on support vector machines for pattern recognition," *Knowl. Discovery Data Mining*, vol. 2, no. 2, pp. 121–167, 1998.
[2] J. Choi, W. J. Jeon, and S. Lee, "Spatio-temporal pyramid matching for sports videos," in *Proc. ACM Int. Conf. Multimedia Inform. Retrieval*, 2008, pp. 291–297.
[3] O. Chum, J. Philbin, M. Isard, and A. Zisserman, "Scalable near identical image and shot detection," in *Proc. Int. Conf. Image Video Retrieval*, 2007, pp. 549–556.
[4] L. Duan, D. Xu, I. W. Tsang, and J. Luo, "Visual event recognition in videos by learning from web data," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit.*, San Francisco, CA, 2010.
[5] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *Proc. IEEE Int. Conf. Comput. Vision*, 2005, pp. 1458–1465.
[6] J. He, S. Chang, and L. Xie, "Fast kernel learning for spatial pyramid matching," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Anchorage, AK, 2008.
[7] P. Jensen and J. Bard, *Operations Research Models and Methods*. New York: Wiley, 2003.
[8] Y. Ke, R. Sukthankar, and L. Huston, "Efficient near duplicate detection and sub-image retrieval," in *Proc. ACM Multimedia Conf.*, 2004, pp. 869–876.
[9] L. Laptev and T. Lindeberg, "Space-time interest points," in *Proc. IEEE Int. Conf. Comput. Vision*, 2003, pp. 432–439.
[10] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Beyond sliding windows: Object localization by efficient subwindow search," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit.*, Anchorage, AK, 2008.
[11] J. Law-To, A. Joly, and N. Boujemaa, *Muscle-VCD-2007: A Live Benchmark for Video Copy Detection*, 2007 [Online]. Available: http://www-rocq.inria.fr/imedia/civr-bench
[12] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features, spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit.*, 2006, pp. 2169–2178.
[13] H. Ling and S. Soatto, "Proximity distribution kernels for geometric context in category recognition," in *Proc. IEEE Int. Conf. Comput. Vision*, Rio de Janeiro, Brazil, 2007.
[14] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, 2004.
[15] J. Munkres, "Algorithms for the assignment and transportation problems," *J. Soc. Ind. Appl. Math.*, vol. 5, no. 1, pp. 32–38, Mar. 1957.
[16] O. Pele and M. Werman, "Fast and robust Earth mover's distances," in *Proc. Int. Conf. Comput. Vision*, Kyoto, Japan, 2009.
[17] Y. Rubner, C. Tomasi, and L. Guibas, "The Earth mover's distance as a metric for image retrieval," *Int. J. Comput. Vision*, vol. 40, no. 2, pp. 99–121, Nov. 2000.
[18] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. IEEE Int. Conf. Comput. Vision*, 2003.
[19] F. Tang and Y. Gao, "Fast near duplicate detection for personal image collections," in *Proc. ACM Int. Conf. Multimedia*, 2009, pp. 701–704.
[20] TRECVID [Online]. Available: http://www-nlpir.nist.gov/projects/trecvid
[21] K. Vaiapury, P. K. Atrey, M. S. Kankanhalli, and K. Ramakrishnan, "Non-identical duplicate video detection using the sift method," in *Proc. Int. Conf. Visual Inform. Eng.*, 2006, pp. 537–542.
[22] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer, 1995.
[23] Y. Wang, J. Ostermann, and Y. Zhang, *Video Processing and Communications*. Englewood Cliffs, NJ: Prentice Hall, 2001.
[24] X. Wu, A. Hauptmann, and C. Ngo, "Practical elimination of near duplicates from web video search," in *Proc. ACM Multimedia Conf.*, 2007, pp. 218–227.

[25] Z. Wu, S. Jiang, and Q. Huang, "Near-duplicate video matching with transformation recognition," in *Proc. ACM Multimedia Conf.*, 2009, pp. 549–552.
[26] D. Xu and S. Chang, "Visual event recognition in news video using kernel methods with multilevel temporal alignment," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit.*, Minneapolis, MN, 2007.
[27] D. Xu and S. Chang, "Video event recognition using kernel methods with multilevel temporal alignment," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1985–1997, Nov. 2008.
[28] D. Xu, T. J. Cham, S. Yan, and S. Chang, "Near duplicate image identification with spatially aligned pyramid matching," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit.*, Anchorage, AK, 2008.
[29] D. Zhang and S. Chang, "Detecting image near duplicate by stochastic attribute relational graph matching with learning," in *Proc. ACM Multimedia Conf.*, 2004, pp. 877–884.
[30] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *Int. J. Comput. Vision*, vol. 73, no. 2, pp. 213–238, Jun. 2007.
[31] W. Zhao, C. Ngo, H. Tan, and X. Wu, "Near duplicate keyframe identification with interest point matching and pattern learning," *IEEE Trans. Multimedia*, vol. 9, no. 5, pp. 1037–1048, Aug. 2007.

Dong Xu (M'07) received the B.Eng. and Ph.D. degrees from the Electronic Engineering and Information Science Department, University of Science and Technology of China, Hefei, China, in 2001 and 2005, respectively. During his Ph.D. studies, he was with Microsoft Research Asia, Beijing, China, and the Chinese University of Hong Kong, Hong Kong.

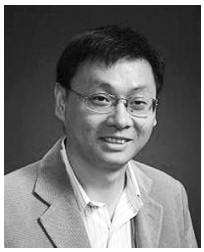
He spent one year at Columbia University, New York, as a Post-Doctoral Research Scientist. He is currently an Assistant Professor with the School of Computer Engineering, Nanyang Technological University, Singapore. His current research interests include computer vision, statistical learning, and multimedia content analysis.

Dr. Xu was co-author (with his Ph.D. student Lixin Duan) of a paper that won the Best Student Paper Award in the prestigious IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) in 2010.

Tat Jen Cham received the B.A. (Engineering) and the Ph.D. degrees from the University of Cambridge, Cambridge, MA, in 1993 and 1996, respectively.

He was a Jesus College Research Fellow with the University of Cambridge from 1996 to 1997, before joining DEC/Compaq Research Laboratory, Boston, MA, as a Research Scientist during 1998–2001. While at Nanyang Technological University, Singapore, he was concurrently a Faculty Fellow in the Singapore-MIT Alliance Computer Science Program from 2003 to 2006. He holds eight U.S. patents and filings. He is currently an Associate Professor and Director of the Center for Multimedia and Network Technology, School of Computer Engineering, Nanyang Technological University, Singapore.

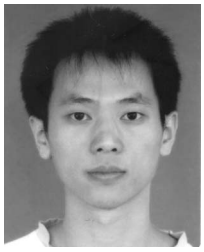
Dr. Cham has made key contributions in computer vision and projector-camera systems, receiving overall Best Paper Prizes at PROCAMS'05, BMVC'94, and in particular at ECCV'96. He is on the editorial board for the International Journal of Computer Vision and an Area Chair for ICCV'09. He has been interviewed on national media in relation to SMA projects that he jointly supervised.



Shuicheng Yan (M'06–SM'09) received the Ph.D. degree from the School of Mathematical Sciences, Peking University, Beijing, China, in 2004.

He spent three years as a Post-Doctoral Fellow with the Chinese University of Hong Kong, Hong Kong, and then with the University of Illinois, Champaign. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. He has authored or co-authored over 150 technical papers over a wide range of research topics. In recent years, his current research interests have focused on computer vision (biometrics, surveillance, and internet vision), multimedia (video event analysis, image annotation, and media search), machine learning (feature extraction, sparsity/non-negativity analysis, and large-scale machine learning), and medical image analysis.

Dr. Yan has served on the editorial board of the *International Journal of Computer Mathematics*, has served as a Guest Editor of the special issue for *Pattern Recognition Letters*, and has been serving as the Guest Editor of the special issue for *Computer Vision and Image Understanding*. He has served as a Co-Chair of the IEEE International Workshop on Video-Oriented Object and Event Classification (VOEC'09) held in conjunction with ICCV'09. He is the Special Session Chair of the Pacific-Rim Symposium on Image and Video Technology, in 2010. He is an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.



Lixin Duan received the B.Eng. degree from the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, China, in 2008. He is currently pursuing the Ph.D. degree from the School of Computer Engineering, Nanyang Technological University, Singapore.

His current research interests include computer vision and machine learning.

Shih-Fu Chang (F'04) leads the Digital Video and Multimedia Laboratory, Department of Electrical Engineering, Columbia University, where he conducts research in multimedia content analysis, image/video search, multimedia forgery detection, and biomolecular image informatics. Systems developed by his group have been widely used, including VisualSEEk, VideoQ, WebSEEk for visual search, TrustFoto for online image authentication, and WebClip for video editing. His group has made significant contributions to the development of MPEG-7 international multimedia standard.

Mr. Chang was the Editor-in-Chief of the IEEE SIGNAL PROCESSING MAGAZINE from 2006 to 2008, a recipient of the Navy Office of Naval Research Young Investigator Award, the IBM Faculty Development Award, and the NSF CAREER Award. His group has received several Best Paper or Student Paper Awards from IEEE, the Association for Computing Machinery (ACM), and the International Society for Optical Engineers. He worked in different capacities in several media technology companies and served as a General Co-Chair for the ACM Multimedia Conference in 2000 and the IEEE International Conference on Multimedia and Expo in 2004.