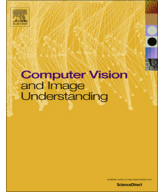




Contents lists available at ScienceDirect

# Computer Vision and Image Understanding

journal homepage: [www.elsevier.com/locate/cviu](http://www.elsevier.com/locate/cviu)

## Two dimensional hashing for visual tracking <sup>☆</sup>

Chao Ma, Chuancai Liu <sup>\*</sup>

School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, PR China



### ARTICLE INFO

#### Article history:

Received 27 August 2014

Accepted 16 January 2015

Available online 30 January 2015

#### Keywords:

Hashing

Tracking

Appearance model

Incremental learning

### ABSTRACT

Appearance model is a key part of tracking algorithms. To attain robustness, many complex appearance models are proposed to capture discriminative information of object. However, such models are difficult to maintain accurately and efficiently. In this paper, we observe that hashing techniques can be used to represent object by compact binary code which is efficient for processing. However, during tracking, online updating hash functions is still inefficient with large number of samples. To deal with this bottleneck, a novel hashing method called two dimensional hashing is proposed. In our tracker, samples and templates are hashed to binary matrices, and the hamming distance is used to measure confidence of candidate samples. In addition, the designed incremental learning model is applied to update hash functions for both adapting situation change and saving training time. Experiments on our tracker and other eight state-of-the-art trackers demonstrate that the proposed algorithm is more robust in dealing with various types of scenarios.

© 2015 Elsevier Inc. All rights reserved.

### 1. Introduction

Visual tracking, as a main topic in computer vision, has been widely used in many applications, such as augmented reality, surveillance and object identification. Numerous tracking algorithms have been proposed during past decades [1–19]. Most visual trackers can be divided into four parts: feature extraction, appearance model, sampling method and online learning model. Among these four parts, appearance model is the key element to the success of coping with occlusion, pose change, illumination and other complex scenarios.

How to design an effective and robust appearance model is the main task in most state-of-the-art trackers. These algorithms often construct appearance models by low-level and mid-level cues such as: Raw pixel, Color Histogram, Gradient, Superpixel, Texture, Haar-like, SIFT, SURF, MSER, LBP, HoG, Harris corner and Saliency object [20]. Mei and Ling [1] treat tracking as sparse approximation problem, which uses raw pixel representation. Based on this idea, much works [2–6] focus on using sparse representation method to solve the tracking problem and have achieved good performance. He et al. [7] propose a new histogram extraction method called local sensitive histogram. Based on this histogram technique, illumination invariant feature is proposed, which is proven to be effective to track object. Wang et al. [8,9] propose superpixel

tracking algorithm, which uses superpixels to generate local parts' confidence of object. The tracker is proven to be effective to occlusion and pose change by its unique appearance model. Some other work [10,11] also use superpixel feature to construct robust appearance model for visual tracking and achieve well tracking results in experiments. Chen et al. [12] integrate several types of features and construct a robust appearance model. The model is combined with complex cells by adaptive weights and appears to be effective to deal with occlusion, pose change and illumination. Besides these trackers, there are also many other algorithms [13–15,17–19], which use the artificial feature or model as the base of tracking algorithm.

An alternative method for constructing appearance model is data dependent technique. Given sufficient descriptor or raw pixel of samples, the data dependent method aims to find a map from high dimensional space to low dimensional space. The most representative literature based on this method is IVT [21,22]. Given the samples at first frame, the tracker obtains a low dimensional representation by PCA, and candidates obtained from coming frames would be mapped into low dimensional feature space to get compact representation. By the support of theory, IVT achieves well performance in experiments. However, the tracker does not take semantic information into consideration and the mapping functions have limited discriminability, which may cause drift or loss of target under complex cases. Compressive tracking [23,24] is another work that uses dimensionality reduction manner. It uses compressive method to reduce the dimension of original feature vector. Although the tracker owns the ability to deal with some

<sup>☆</sup> This paper has been recommended for acceptance by Y. Aloimonos.

<sup>\*</sup> Corresponding author.

E-mail addresses: [njustmachao@163.com](mailto:njustmachao@163.com) (C. Ma), [chcailiu@163.com](mailto:chcailiu@163.com) (C. Liu).

tracking situations, its performance is instable because of the random process in mapping method. Recently data-dependent hashing approach becomes a popular method in large-scale vision problems. It is shown to be a good method to expedite similarity computation. This property makes it possible to apply the hashing method to tracking algorithms. Li et al. [25] propose learning compact binary code for visual tracking. The algorithm always keeps a sample buffer and updates the hash functions every ten frames. Although the tracker obtains effective tracking results in experiments, its training stage is not an efficient manner to update hash functions.

To better apply hashing method in tracking and provide an efficient updating strategy, we proposed two dimensional hashing method for visual tracking. Inspired by 2DPCA [26] and 2DLDA [27], the proposed two dimensional hashing method is based on 2D sample matrices (object image) rather than 1D feature vectors. The algorithm does not need to vectorize the sample matrices, and then the dimension of covariance matrix is reduced dramatically, which is the key element for computational efficiency. Given training samples at first frame, the hash functions are initialized by the proposed hash method. Then we can formulate the binary codes of candidates at coming frame. Confidences of candidates are calculated by their hamming distances to the templates, and the highest one is chosen as target at current frame. For adapting to the situation changing, hash functions are updated by the designed incremental learning model during tracking. Due to this learning model, our algorithm only needs to store few matrices. Experimental results show that the approach achieves good performance in evaluation metrics CLE, SR and FPS, which demonstrate the effectiveness and efficiency of our tracker, and outperforms the state-of-art tracking algorithms.

The rest of our paper is organized as follows: Section 2 introduces some related literatures. Section 3 presents the details of two dimensional hashing method. Section 4 shows the framework of tracking by two dimensional hashing. Experimental results are analyzed detailedly in Section 5. And final conclusion is drawn in Section 6.

## 2. Related work

Recently, hashing techniques attract great attention in computer vision and pattern recognition due to its efficiency and accuracy in data organization. Hashing method aims to map the original high-dimensional data into a compact binary code, while preserving structure of original data. After mapping to binary code, the pairwise comparison could be calculated efficiently. Hashing techniques can be categorized to data-independent methods [28–32] and data-dependent methods [33–40,25]. One of the most well-know data-independent hashing method is Local Sensitive Hashing (LSH) [28], which generates hash code by random projections. Based on LSH's idea, many methods [29–32] belonging to the LSH family have been proposed and have achieved good performance in computer vision and pattern recognition applications. Compared to data-independent techniques, more recent works have focused on data-dependent techniques with the goal of capturing more compact binary codes. They learn hash functions by optimizing objective function, which preserves samples similarity in Hamming space. However, directly solving optimization problem is very difficult. Thus relaxation technique is often used to simplify the optimization. Spectral hashing [33] uses spectral relaxation method to solve this problem and has proved successful. Liu et al. [34] propose hashing with graph, which utilizes manifold structure to generate binary codes of data. The widely developing deep learning techniques are also applied into hashing techniques [41–43]. They use common framework (RBM and neural network) of deep learning to learn the binary code of data, which achieves

good performance in computer vision applications. Recently LDA Hashing [35] and Semi-supervised Hashing [36] are proposed. They both provide methods to learn hash functions with labeled train samples. With the success of hashing methods, many applications [37,44] begin to use hashing technique to solve the real problems.

Incremental learning is widely used to online learning applications, which is suit for objecting tracking. Many incremental learning works [27,45–49] have been proposed and applied successfully. Pang et al. [46] propose an Incremental LDA for online face recognition. They address the method of updating the scatter matrices with incremental manner, but the updating scheme is still time and memory consuming. To solve speed limitation, Kim et al. [47] use sufficient spanning set approximation to address problem of incremental LDA. The method significantly improves the calculation speed and provides the similar results with batch LDA. Based on 2DLDA, Wang et al. [48] propose incremental 2DLDA for face recognition, which achieves fast speed and effective performance. And the idea is also applied to the tracking algorithm by Li et al. [45]. Zhao and Yuen [27] improve the LDA/GSVD algorithm and fast SVD updating technique by the proposed incremental supervised learning method called GSVD-ILDA which can incrementally learn an adaptive subspace instead of recomputing the LDA/GSVD. Lu et al. [49] extend the CLDA algorithm to online manner by incremental method and apply it to face recognition successfully.

In our work, we aim to apply data-dependent hashing technique into object tracking, which is challenge. The hashing methods mentioned above are designed for off-line training application which is not suit for online training. If we use batch manner to apply these hash functions directly in tracking, it will take several seconds per-frame. Even if it may achieve good performance, the processing time could not be accepted. So in order to solve this problem, we use incremental learning model to update our hashing functions at each frame. However, because of the demands of hashing method, the covariance matrices are not same as other incremental LDA algorithms such as [45], which makes original incremental method cannot be applied directly. Fortunately, by the proof of Theorem 1, we could update the covariance matrices incrementally and improve the tracking speed. Overall, we list our main contributions of our tracker as below:

1. In this paper, we propose a 2D-based hashing method which could fast extract the binary feature of samples.
2. We successfully apply the hashing method into tracking model by some details.
3. We design an effective and suitable learning model to update hash functions at every frame.
4. Comparison experiments are done to demonstrate the effectiveness and efficiency of our tracker.

## 3. Two dimensional hashing

In this section, we introduce the details of our hashing method. Given training image samples  $\mathcal{X} = \{\mathbf{A}_i, y_i\}, i = 1, \dots, n$ ,  $\mathbf{A} \in \mathbb{R}^{D \times D}$ ,  $y_i$  is the label of  $i$ -th sample, which represents positive (foreground) or negative (background). In these samples, pairs of them are categorized into two sets: neighbor pair set  $\mathcal{NE}$  and non-neighbor pair set  $\mathcal{NO}$ . A pair of  $(\mathbf{A}_i, \mathbf{A}_j) \in \mathcal{NE}$  is represented as neighbor pair (both positive or both negative). Similarly,  $(\mathbf{A}_i, \mathbf{A}_j) \in \mathcal{NO}$  is non-neighbor pair (one of them is positive and the other is negative).  $\{\mathbf{A}_i^c, y_i^c\}_{i=1}^n$  are centralized samples of  $\mathcal{X}$ . So the proposed hashing method aims to map data  $\mathbf{A}_i$  to hamming space to obtain binary representation. Suppose the number of projection directions to be  $K$ , that is to say  $K$  hash functions need to be learned by training samples. Follow general idea, our hash function is defined as

Download English Version:

<https://daneshyari.com/en/article/525560>

Download Persian Version:

<https://daneshyari.com/article/525560>

[Daneshyari.com](https://daneshyari.com)