ISSN: 2321-7782 (Online)
Volume 3, Issue 7, July 2015
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

Duplicate Video Detection by using keypoint Descriptor

Tejaswini Mali[1]
Computer department,
JSPM's JSCOE, Pune
Pune, India

Prof. H. A. Hingoliwala[2]
Assistant Professor, Computer department,
JSPM's JSCOE, Pune
Pune, India

Abstract: Scale Invariant Feature transforms is a concept which is used to detect and describe local features in images. Basically SIFT is not flip invariant. There are many flip or flip like transformations observed in images because of mirror capturing or opposite capturing view or flipping of images. In this paper a new concept or new descriptor proposes flip-invariant SIFT or we can say it as F-SIFT. It preserves original properties of SIFT while it also allows flips and rotates. F-SIFT starts by evaluating major curves of local patch and geometrically normalizes the patch by flipping before computing for SIFT. The advantages of F-SIFT are demonstrated using 3 tasks- large scale copy detection, object recognition and detection. In copy detection, a framework is proposed which collects the flip properties of F-SIFT for rapid filtering and weak geometric checking. F-SIFT improves the detection accuracy of SIFT and also saves more than 50 percent of computational costs. In object recognition, advantages of F-SIFT in dealing with flip transformation by comparing it to seven other descriptors are shown. In object detection, the ability of F-SIFT in describing symmetric objects is demonstrated. We can witness consistent improvement across different kinds of keypoint detectors for F-SIFT over the original SIFT.

Keywords: Scale Invariant Feature Transform, FSIFT , Object Detection, Object recognition.

I. INTRODUCTION

The attractiveness of SIFT is mainly due to its invariance to various image transformations including: rotation, scaling, lighting changes and displacements of pixels in a local region. SIFT is normally computed over a local salient region which is located by multi-scale detection and rotated to its dominant orientation. As a result, the descriptor is invariant to both scale and rotation. Furthermore, due to spatial partitioning and 2D directional gradient binning, SIFT is insensitive to color, lighting and small pixel displacement. Despite these desirable properties, SIFT is not flip invariant. Goal of the proposed application to detection copy video and object detection. And generate F-SIFT which enhance the properties of the SIFT. Because of the success of SIFT, image local features have been extensively employed in a variety of computer vision and image processing applications. Particularly, various recent works take advantage of SIFT to develop advanced object classifiers. There we proposed an application F-SIFT which enhance SIFT with flip invariance property. The employment of F-SIFT for video copy detection, object recognition and detection is also demonstrated. Particularly, we show that, by smartly indexing F-SIFT, the performance improvement in both detection accuracy and speed could generally be expected

II. LITERATURE SURVEY

While developing local descriptors invariant to various geometric transformations has received numerous research attention, the property of flip invariance surprisingly is often not considered. Until recently, there are several flip invariant descriptors including RIFT,SPIN, MI-SIFT and FIND .These descriptors, as well as SIFT, mainly differ by the partitioning scheme of narrow region. SIFT, which divides a region into 4 4 blocks and describes each grid with an 8 directional gradient histogram, generates the feature by concatenating the histograms in row major order from left to right and the histogram bins in clockwise manner. As a result, flip transformation of the region will disorder the placement of blocks and bins. This results in a dissimilar version of descriptor due to the predefined order of feature scanning. The potential solutions for dealing with this difficulty include altering the partitioning scheme or scanning order and feature transformation. RIFT adopts a different partitioning scheme than SIFT by dividing a region along the log-polar direction. Similar to SIFT, the 8-directional histograms are computed for each division and then concatenated to form a descriptor. Since the partitioning scheme itself is flip and rotation invariant, RIFT is not sensitive to order of scanning. On the other hand, while this radius based Partition schemes of (a) SIFT (b) RIFT (c) GLOH (d) SPIN and (e) FIND . division is smooth and less vulnerable to quantization loss if compared to grid-based partitioning, the spatially loose representation also results in RIFT a descriptor not as distinctive as SIFT. GLOH which can be viewed as an integrated version of SIFT and RIFT provides finer partitioning. FIND is a new descriptor which allows overlapped partitioning and scans the 8 directional gradient histograms by following the order indicated. Under this scheme, the descriptors produced before and after a flip operation are also mirror of each other. Specifically, a descriptor generated as a result of ip can be recovered by scanning the histograms in reverse order. With this property, FIND explicitly makes the property of the descriptor invariant to flip by estimating whether a region is left or right pointing through parameter thresholding. When comparing two descriptors of left and right pointing respectively, the descriptor components are rearranged on the y for proper order of feature matching. Nevertheless, as reported in , the estimation of pointing direction is highly dependent onparameter setting, and more importantly, incorrect estimation directly implies invalid matching result. In addition, similar to RIFT, the partitioning scheme does not produce descriptor as distinctive as SIFT. MI-SIFT , instead, operate straight on SIFT while transform it to a new descriptor which is flip invariant. This is achieved by explicitly identifying the groups of feature mechanism which are disorderly located as a result of flip operation. MI-SIFT labels 32 of such groups and represents each group with four moments which are flip invariant. Nevertheless, the descriptor based on moment is not discriminative.The problem of flipped copy detection is engineered by indexing two SIFT descriptors for each region of which one of them is computed by simulating flip operation This results in significant increase in both indexing time and memory consumptions. An another strategy was working by submitting two versions of descriptors, flipped and without flipped, as query for copy detection. This strategy introduces the disadvantage that the query processing time is double. Most of the key point detectors and visual descriptors are proposed for feature point matching in object recognition. Dissimilar from copy detection and object recognition, the existing works on object detection are mostly learning based. Specifically, bag-of- visual-words (BoW) constructed from local features such as SIFT are input for classifier le. To the best of our awareness, no work has yet seriously addressed the issue of detection performance by contrasting features with and without incorporating flip invariance property

III. PRELIMINARY WORK (PHASE-I)

Scale-invariant feature transform (SIFT) feature has been widely accepted as an effective local keypoint descriptor for its invariance to rotation, scale, and lighting changes in images. we proposed a new descriptor, named flip-invariant SIFT preserves the original properties of SIFT while being tolerant to flips. The proposed system used for video copy detection and object detection. This system is also useful for flipped and rotated video detection. By using the F-SIFT the performance improved in both detection accuracy and speed. In first module keypoint descriptor is used to find out the particular point from images. Simply it can be said as those are used for extracting the image features. These keypoints with rotedand flipped images are shown in figure 1 and figure 2.

Figure 1. Roted Image with keypoint descriptor

Figure 2.Flipped image with keypoint descriptor.

As images are shown with keypoint descriptor, same concept is used to implement video copy detection. To start implementation we have to start with describing and locating Keypoint descriptor. These keypoint detectors are indeed flip invariant and capable of locating regions under various transformations. There are two factors used for locating key points one is Flip Invariant Detectors and another is FSIFT descriptors.

1. Flip invariant detector:-Analysis on flip invariance of major detectors is given as follows.

Given a pixel *P*, the second moment matrix is defined to describe gradient distribution in the local neighbourhood of *P*:

$$\mu(P,\sigma_I,\sigma_D)=\sigma_D^2 g(\sigma_I)*\begin{bmatrix} L_x^2(P,\sigma_D) & L_xL_y(P,\sigma_D) \\ L_xL_y(P,\sigma_D) & L_y^2(P,\sigma_D) \end{bmatrix}$$

Where σI is the integration scale, σD is the differential scale and Lg is to compute the derivative of P in g (x or y) direction. The local derivatives are computed with Gaussian kernels of the size determined by the scale σD. FSIFT Descriptor :- While keypoint detectors are mainly flip invariant, there is no guarantee that the features extracted from salient regions are also flip invariant. As discussed earlier, the invariance is mainly dependent relative on the layout of partitioning scheme in a descriptor. Different from the existing approaches, our aim here is to enrich SIFT to be flip invariant while preserving its creative properties including the grid-based quantization. Flip transformation can happen next to arbitrary axis. However, it is easy to imagine that any flip can be decomposed into as a flip along a predefined axis followed by a certain degree of rotation as shown in Figure 3. Thus, an intuitive idea to make a descriptor flip invariant is by normalizing a local region before feature extraction through rotating the region to a predefined axis and then flipping it along the axis. Furthermore, if a region has been rotated to its dominant orientation which is the case for regions identified by keypoint detectors, the normalization can be just done by flipping the region horizontally (or vertically). In other words, a prominent solution for flip invariance is to determine whether flip should be performed before extracting local feature from the region. We propose dominant curl computation to answer this question. Curl is mathematically defined as a vector operator that describes the infinitesimal rotation of a vector field. The direction of curl is the axis of rotation determined by the right-hand rule. In multivariate calculus, given a vector field *F(x, y, z)* defined in *R*3 which is differentiable in a region, the curl of *F* is given by

$$\nabla \times F = \begin{vmatrix} i & j & k \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ F_1 & F_2 & F_3 \end{vmatrix}.$$

According to Stokes' theorem, the integration of curl in a vector field can be expressed by

$$\iint_{\Sigma \in R^3} = \nabla \times F \cdot d\Sigma.$$

In our case, curl is defined in a 2D discrete vector field *I*. The curl at a point is the cross product on the first order partial derivatives along *x* and *y* directions respectively. The flow (or dominant curl) along the tangent direction can be defined by

$$C = \sum_{(x,y)\in I} \sqrt{\frac{\partial I(x,y)}{\partial x}^2 + \frac{\partial I(x,y)}{\partial y}^2} \times \cos\theta$$

Where

$$\frac{\partial I(x,y)}{\partial x} = I(x-1,y) - I(x+1,y)$$
$$\frac{\partial I(x,y)}{\partial y} = I(x,y-1) - I(x,y+1)$$

And θ is the angle from direction of the gradient vector to the tangent of the circle passing through *(x, y)*.

Generally, there are only two possible directions for *C*, either clockwise or counter clockwise, which is indicated by its sign. The sign changes only when the vector field has been flipped (along an arbitrary axis). If we enforce every local region that the sign of flow is clockwise, the normalization is performed by flipping the regions whose signs are counter clockwise. In other words, the solution for whether to flip a region prior to feature extraction is based on the sign of *C*.

For robustness, Equation can be further enhanced by assigning higher weights to vectors closer to region center as following

$$C = \sum_{(x,y)\in I} \sqrt{\frac{\partial I(x,y)}{\partial x}^2 + \frac{\partial I(x,y)}{\partial y}^2} \times \cos\theta \times G(x,y,\sigma)$$

Where the flow is weighted by a Gaussian kernel *G* of size σ equal to the radius of local region1.

To summarize, F-SIFT generates descriptors as following. Given a region rotated to its dominant orientation, Equation is computed to estimate the flow direction of either clockwise or anticlockwise

IV. Goals and Objectives

The Objective of the proposed application are as follows:

- Video Copy Detection
- Object detection
- Object recognition.

Technology used for implementing the system is java and for java , Netbeans and opencv is used.

V. Implementation Constraint (Phase II)

1. Feasibility Study

The feasibility study comprise of an initial investigation into personnel will be required. Feasibility study will help you make informed and transparent decisions at crucial points during the developmental process. All projects are feasible given unlimited times and resources. Unfortunately, the development of computer based system is more likely to be plagued to scarcity of resources .It is both necessary and prudent to evaluate the feasibility of project at earliest possible time.

2. Technical Feasibility

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed. Technical issue raised during the investigation is,The project should be developed such that the necessary functions and performance are achieved within the constraints. The project is developed within latest technology.

3. Overview of Implemented System

The modules of the implemented application are as follows:

- Input Video: For the training phase user give the input video to the proposed system.
- Frame Conversion: Input videos are converted into frames. For the Feature extraction process.
- Feature Extraction: From the input frame the features are extracted. The features are extracted in the form of SIFT. The object detection is done on this phase.
- Database: After the feature extraction, the keyframes are extracted. This keyframes are saved in the database for the detecting the copy video.
- Query Video: The query video is uploaded by the user. After uploading this query video the keyframes of the query video are extracted.
- Matching: The matching process of the keyframes of the query video and the input video is done. If the keyframes are matches then the query video is detected as a copy video. Otherwise the keyframes of the query video is saved in the database

4. System Architecture

System architecture of video copy detection is shown in following figure

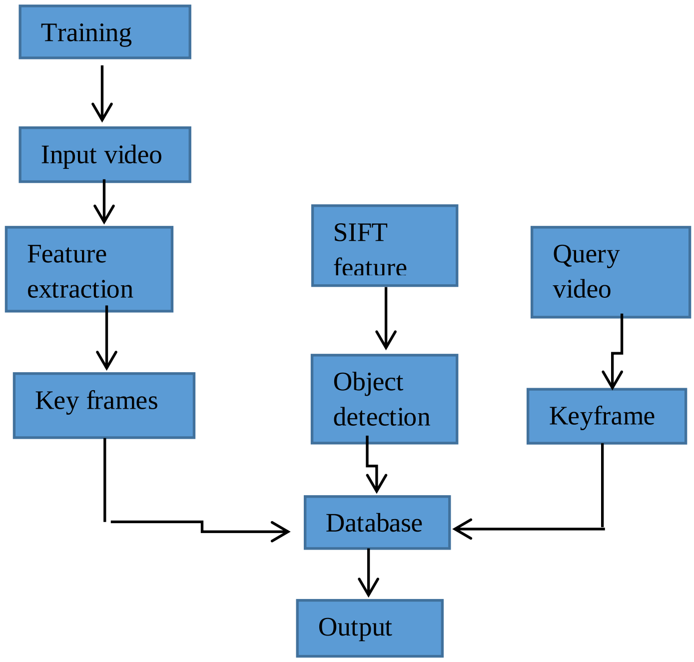

Figure.3 System Architecture of System

5. Mathematical Model For System

$$sim(Q,R) = \frac{\sum h(q,p)}{(\| BoW(Q\|2)\|(BoW(R\|2)}$$

6. Working of System

Whenever input query(Q) is given to the system , It will compare input query with the database video(R). Firstly system creates the frames of input query and database query. Then in first search means without key frame it checks one to one frame matching and in another means in key frame extraction it compares random frames. Because of this time required for video matching is less than one to one. For this we extract image features from video and then comparison is done. Graph show exactly how much time is reduced during one to one frame checking and random frame checking.

7. Graph of Result

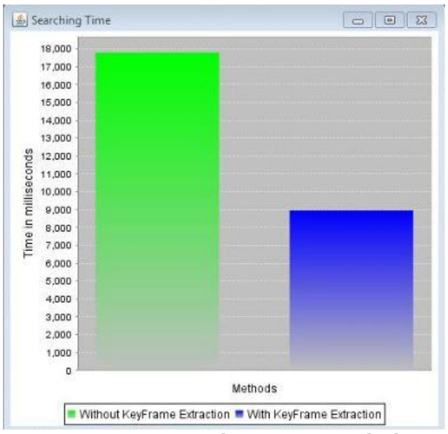

Figure 4 Comparison between two methods

VI. CONCLUSION

We have presented F-SIFT and its utilization for video copy detection, object recognition and image classification. On one hand, the extraction of F-SIFT is slower than SIFT due to the computation of dominant curl and explicit flipping of local region. On the other hand, the improvement in detection effectiveness is consistently observed in three applications. Video copy detection, in particular, demonstrates significant improvement in recall and precision with the use of F-SIFT. More importantly, by wisely indexing the F-SIFT with extra overhead of one bit per descriptor in space complexity, the speed of online detection on a dataset of 0.9 thousand keyframes has also been improved by about two times. This indeed has compensated the need for longer time in feature extraction.

References

1. F. Perronnin and C. Dance “Fisher kernels on visual vocabularies for image categorization” in Proc. Int. Conf. Comput. Vis. Pattern Recognit., Jun. 2007, pp. 18
2. M.-C. Yeh and K.-T. Cheng “A compact, efective descriptor for video copy detection" in Proc. Int. Conf.Multimedia, 2009, pp. 633636.
3. F. Perronnin, J. Sanchez, and T. Mensink, Improving the Fisher kernel for large-scale image classification” in Proc. Eur. Conf. Comput. Vis., 2010, pp. 143156.
4. Z. Liu, T. Liu, D. Gibbon, and B. Shahraray “Effective and scalable video copy detection” in Proc. Int.Conf. Multimedia Inf. Retr., 2010, pp. 119128.
5. M. Douze, H. Jgou, and C. Schmid “An image-based approach to video copy detection with spatio-temporal” post-_ltering in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2006, pp.21692178.