Received: May 29, 2019

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Adaptive Near Duplicate Image Retrieval Using SURF and CNN Features

Tejas Mehta[1]* **Chandu Bhensdadia[2]**

[1]*Charotar University of Science and Technology, India*
[2]*Dharmsinh Desai University, India*
* Corresponding author's Email: tejsmehta@gmail.com

Abstract: In this paper, we present an adaptive approach in order to match and retrieve near duplicate images at different scales. Matching only local Features does not necessarily identify visually similar images. Global features are fast at matching but less accurate. Many existing methods either use local features or CNN features for image or video retrieval task. In this paper, we combined the use of SURF local points and CNN features extracted around SURF points in order to match near duplicate image pairs. Image pairs are segmented into blocks and CNN features of the image block containing matched SURF features are extracted and matched. Regions around matched image blocks are grown adaptively and matching is carried out until CNN mismatch is observed. To verify our proposed approach, experiments are carried out on benchmarking California-ND and Holiday dataset. Compared to traditional approaches for image retrieval, our approach not only retrieves relevant images but also provides detail of localized matched patch. For California-ND dataset and Holiday dataset, we achieve remarkable mAP (mean average precision) score up to 0.86 and 0.74 respectively.

Keywords: CNN (convolution neural network) features, VGG19 (visual geometry group), Near duplicate image retrieval, Adaptive matching.

1. Introduction

Matching image for similarity has got remarkable attention for variety of computer vision tasks. With the rapid growth of internet and social media, volume of multimedia content uploaded is growing substantially. Many among them are found to be near duplicates. Near duplicate images/videos are identical or near identical images/videos acquired from different camera or change in viewpoints, different lightning conditions, undergone with various editing operations such as addition, deletion or content modification, different foreground or background objects etc. Detecting near duplicate image is a fundamental task for various computer vision applications. Copyright infringement is found to be an important application of detecting image near duplicate. However here it should be noted that every near duplicate image need not to be the copy of original image with some modification to detect image copy. Near duplicate identification is broadly categorized as near duplicate retrieval or near duplicate detection [1]. Near duplicate retrieval involves retrieving all images or videos based on query input while detection involves detecting near duplicates among all pairs of image or videos. Discovering duplicate image clusters is found to be one of the efficient ways in case of detecting image near duplicates [2]. Majority of the research focuses on visual similarity to detect ND (near duplicate) pairs. It is important to note that in order to identify near duplicate image or video, matching visual similarity of image/key-frame pairs is an important task. Finding near duplicate contents are common in web, TV news broadcasting etc. According to a survey, nearly 22% images retrieved from the web search are near duplicates. Near duplicate contents are not limited to image/video. Many duplicate or near duplicate documents exist on web. Search results are affected due to such redundancies exists among documents on the web [3]. In some cases focus may be to

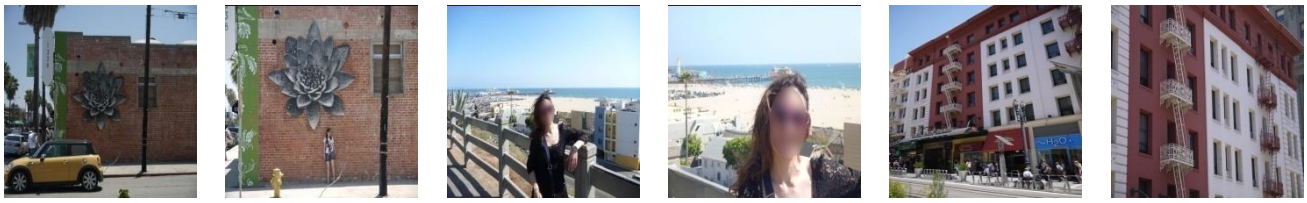


Figure. 1 Left pair Near Duplicate image pair with different foreground objects. The Middle pair is Near duplicate with different zooming condition. The right image pair is extreme example of near duplicate with change in view point (From California ND dataset)

retrieve similar contents while in other cases focus may be to filter out redundant or near redundant contents. Our focus in this paper is to retrieve similar (duplicate/near duplicate) images. There are various difficult to detect cases of 'image near duplicates' such as severe zooming, change in viewpoint, different exposures, viewpoint and background change, burst shots and other combination of various cases. Fig. 1 illustrates some cases of near duplicate image pairs.

Traditionally local or global features are extracted and matched to detect whether given image pair is near duplicate or not. Local features such as SIFT (scale-invariant feature transform) or SURF (speeded up robust features) have been widely used to detect similarity of image pair. However these features may give wrong matches. In our approach, not only local features are matched but regions surrounding the local features are matched in order to filter out wrong matches.

Matching neighbour region helps us to consider the portion of image that does not contain any local features. We used combined power of CNN [4] and SURF [5] features to achieve our goal. SURF features provide fast and reasonable local match while CNN feature provides powerful and robust categorical match. We do not compute CNN features in advance instead CNN features are computed on the fly for the region surrounded local feature and set of matched images are returned without extensive storage of image descriptors of each image. In our approach, CNN training is not carried out as our aim is to match any two image pairs irrespective of the data set. We tested our approach using popular deep learning architecture VGG19 [6]. The contribution of this paper is summarized as follows.

1) Adaptive matching: We used matching strategy in adaptive manner resulting in local match to global match. We match local patch with the help of SURF descriptor and then patch window is expanded to obtain maximum coverage of entire image. This approach allows us to match image pairs even though only single local feature match is observed in some cases.

2) Combined use of SURF and CNN features: SURF matching in the first stage helps us to match the same object eliminating the need of training to match the same object. At later stage powerful CNN feature helps us to match image patch with robustness.

3) Coverage and correlation based matching: Traditionally image retrieval techniques do not provide any information about what portion of image is actually matched. Our technique provides results in terms of number of blocks matched for given image pairs as well as correlation for the matched pairs. Results are returned in descending order of number of matched blocks multiplied by correlation value of matched patch. In addition to that location of matched pair can also be obtained. Sample matched locations are shown in Fig. 5.

The paper is organized in various sections. Section 2 discusses about related work carried out using different types of features. Section 3 and 4 represent detail of our work and experimental results respectively`.

2. Related work

From the aspect of features used to retrieve or detect near duplicate image or video we broadly categorize techniques into two approaches 1) conventional feature base approaches 2) CNN feature based approaches.

2.1 Conventional feature based approaches

Many existing methods rely on extracting and matching local descriptors as fundamental task for near duplicate visual content matching. Local descriptors, which are computed around local features, are more distinctive and robust to geometric and photometric changes in the image and have been an ideal choice for many researchers compared to global features such as histograms, etc. Scale invariant feature transform-SIFT [7] is such a widely used descriptor for various image retrieval task including near duplicate image detection [8,10]. PCA SIFT [9] is modified version of SIFT that is more distinct and compact to handle non rigid

distortions. SURF descriptor [11] is used to match image pairs. Only utilizing such local features lacks semantic representation of an image. Regions around local features play an important role in order to obtain correct match. This motivated us to match local features as well as region surrounded local features. Although local features and their corresponding descriptors are robust and accurate, matching large number of local descriptors are time consuming. Visual keywords, also known as bag of words are popular in retrieval task. With this, key-points are quantized into groups and each group represents visual word. However such quantization may lead to false matching. Simply matching local points are not sufficient to conclude about near duplicate image pairs. Low level features forms basis for BOVW ((Bag of visual words) model. However BOVW model lacks accuracy due to quantization and may give false matches. Less retrieval accuracy is observed for BOVW based model as mentioned in result section. More discriminative power can be added to BOW model [12] by encoding features to VLAD [13] (Vector of Locally Aggregated Descriptors) or FV (Fisher vectors) representation. However, it should be noted that FV encoding [14] is high dimensional and dense. Certain technique need to be employed in order to achieve robustness. To improve performance of matching, various techniques are proposed such as hamming embedding, weak geometric consistency (WGC) [15] and soft weighting (SW) [16]. Hamming embedding is one of the well known state of art techniques that generates binary signature to improve keyword matching [15]. However this method requires dictionaries trained on the given dataset while our approach does not need any pre-trained vocabulary. In [17], Pattern entropy (PE) measure is proposed to evaluate spatial coherency patterns in horizontal and vertical directions to detect near duplicate pairs. However it is observed that PE fails under certain circumstances such as object zooming, change in illumination etc. Pattern coherency measure [18] is proposed in order to handle matching of extreme zooming condition. It should be noted here that our technique handles matching in adaptive manner and is robust to various cases like object zooming, object occlusion etc. by utilizing the power of CNN. In [19], an attribute relation graph (ARG) between interest points is constructed and then similarities of image pair are detected by stochastic attributed graph matching. This method needs heuristic learning parameters as compared to our approach which does not need any learning parameter. Conventional features discussed in this section do not directly incorporate any high level semantics in order to match and retrieve images. Our approach relies on CNN which is found to be effective in order to extract high level features which in turn helps us to retrieve images based on semantic context.

2.2 CNN feature based approaches

Convolution Neural Network (CNN) provides generic feature extractor for image retrieval. Activation of higher layer (normally fully connected layer) of CNN is used in order to obtain generic features. Features extracted from fully connected layer gives best result in image retrieval [20]. Vectors generated at intermediate levels of CNN can be utilized in various image retrieval tasks. CNN features can be obtained for the entire image or at object or patch level in order to retrieve similar images. Applying CNN to the entire image and vectors generated from intermediate CNN levels can be utilized in image retrieval [21]. However CNN feature obtained from the entire image as input is lacking geometric invariance. To improve invariance of CNN activation, Gong obtained patches of different size with stride of 32 pixel and extracted CNN features for patches and results are concatenated [22]. In [23] CNN features at patch level are extracted and aggregated in an order less manner to obtain invariant representation. Object proposals are used in order to detect objects and CNN features are extracted for object level. Object or Patch level CNN features provides better retrieval accuracy [24]. Image pairs with non-rigid deformation and weakly textured regions are matched using a strategy named deepMatching [25] to robustly determine correspondences between two images to match.

In [26], CNN features are extracted of each region generated by the object proposal [27] method for object level features and fused it with CNN features at scene level and with point level SIFT feature for image retrieval. Conventional features like SIFT and deep features like CNN are not alternative to each other. It is observed that they can be used as complimentary to each other [26]. Performance of CNN is found remarkable in various computer vision tasks however it cannot be concluded that CNN is always a better choice than SIFT [28]. Similar work is carried out by fusing SIFT and CNN at different level [29]. CNN is employed for local and global level. Visual matching is carried out for local, regional and global level. CNN feature used in image search acts as auxiliary cues to the BoW model. True match of the key-point is considered if and only if they are

located in all three levels (local, regional, global). It should be noted here that similarity is estimated by fusing similarity on all three levels. Considering global similarity may result into failure to detect ND (near duplicate) pairs that shares common content in picture in picture form and rest of the image contains different contents. Our approach matches the key point locally and then gradually increases patch window till the portion of image gets match. This provides an adaptive matching strategy rather than concluding similarity directly relying on global similarity. Performance degradation is clearly observed by directly matching CNN features of entire image pair as shown in Fig. 7. Our approach is based on detecting similarity based on coverage of image block matching as well as correlation values obtained a patch level. This gives better retrieval performance that can be observed in result section. The following section gives detail of our proposed approach.

3. Proposed approach

Extractions of powerful features are crucial in image retrieval task. We extracted SURF features and CNN features to match image pairs. At first we perform extraction of SURF local features and then local feature matching is carried out. Our objective is not only to match local feature matching but to match surrounded region as well. Matching surrounded region plays an important role in order to eliminate false match. After matching local features, region surrounding local point is matched using CNN features. Each region of image is given unique identification number. To obtain region identity, image is segmented into fixed size blocks and each block is given unique number in row major order.

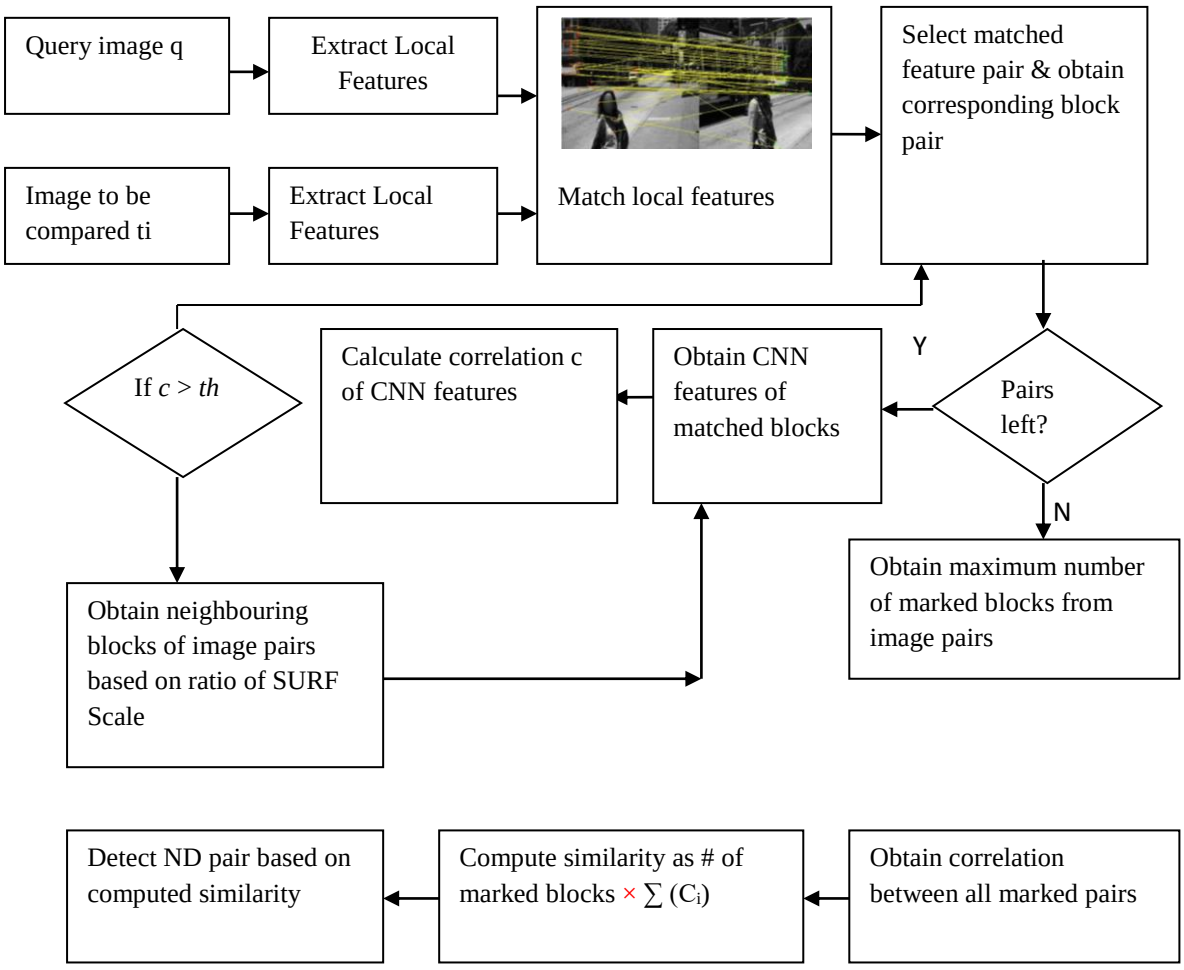


Figure. 2 Process of matching image pair

Followed by SURF local feature matching, pair of blocks containing matched SURF feature points is obtained and CNN features are extracted for corresponding blocks. In order to extract CNN features for the blocks, we used VGG19 [3], a popular pre-trained neural network model. It is currently the most preferred choice in the community for extracting features from images. To extract features, we use activation from fully connected layer 'fc7' (fully connected layer) and obtain 4096 dimensional CNN feature vector. As VGG19 requires input patch size of 224x224x3, each extracted region is resized to 224x224x3. If CNN features of corresponding blocks are correlated then the blocks from image pair is marked so that same block is not considered for further matching. If we get block matching then patch window size is increased and corresponding CNN feature matching is carried out repeatedly until CNN mismatch is observed.

```
Match_pair(q, t_i ∈ imageset)
  Segment image q & t_i into fixed size blocks
  Obtain set of SURF features s1, s2 for q and t_i respectively
  Match set s1 and s2 & get matched point pairs m1 and m2
  For all pairs belongs to m1 do
        Obtain blocks b1 and b2 from image pair q & t_i respectively
        Calculate ratio of scale from matched points m1 and m2
        If block b1 is not marked
            Obtain CNN features f1, f2 for b1 and b2 respectively
            Calculate correlation c1 of f1 and f2.
            Let nbsize be the block size of neighbourhood
            Initialize delta ← 1
            Initialize nbsize← block size
            While c1 > threshold and all blocks from q and t_i are not covered and delta>=0
                If ratio<1 then
                  Obtain r as per Eq. (2)
                  Expand window size of b1 by block size
                  Expand window size of b2 by r times.
                  Obtain CNN features f1' and f2' of expanded window b1 and b2 respectively
                  Calculate correlation c2 of f1' and f2'
                else
                  Obtain r as per Eq. (2)
                  Expand window size of b1 by r times
                  Expand window size of b2 by block size
                  Obtain CNN features f1' and f2' of expanded window b1 and b2 respectively
                  Calculate correlation c2 of f1' and f2'
                end
                delta = c2-c1
                If c2<threshold then exit loop
                Mark blocks covered by b1 & b2 by extracting neighbours mentioned in Fig. 4
                Obtain total number of marked blocks from q and t_i
                Marked_blocks ← max {total number of marked blocks from q,
                                     total number of marked blocks from t_i}
                Increment nbsize by block size
                Update correlation value c1 (c1=c2)
                sum← sum+c1
            end
        end /* block mark*/
  Calculate similarity between image pair as Marked_blocks × sum
```

Figure. 3 Algorithm to match query image with dataset

Extraction of neighbouring blocks depends on ratio of scale of matched SURF features. Ratio of scale of SURF feature helps us to handle matching of image blocks pairs at different zoom levels. Another SURF feature point and its corresponding block regions are obtained for matching only after repeated matching of current block segment is over. Marking process is carried out for all blocks that are found similar from both images.

Marking of blocks helps us to determine the region that has not been considered so far as well as filtering out local feature points located in the same block. When no more SURF points are left, total number of marked blocks is obtained from image pair. This observation leads us to consider maximum number of marked blocks from the image pairs in order to retrieve such pairs as ND (near duplicate) pair. Finally maximum number of marked blocks is obtained to detect similar portion between image pairs. In Fig. 5, it is observed that whole portion of second image is part of first image giving more marked blocks in second image and less in first image. Image pair is considered as ND (near duplicate) pair if sufficient portion of the image is marked. From the obtained results we found that just finding number of blocks for image pairs are not sufficient for efficient retrieval. To improve retrieval performance we incorporated correlation value along with number of matched blocks giving rise to new similarity measure. Similarity measure as shown in Eq. (1) is used in our model where q is query image, t_i is image from the set, n is no. Of times window is expanded C_k is correlation of CNN features for the current patch window; Nm is number of matched blocks.

$$sim(q, ti) = \sum_{k=0}^{n} C_k \times Nm \quad (1)$$

Query image is matched with rest of the image set in order to retrieve set of ND (near duplicate) images. Finally images are sorted in descending values of similarity.

The entire process of image pair matching is shown in Fig. 2 and detailed algorithm is shown in Fig. 3 followed by discussion of extraction of neighbouring blocks. Output of the algorithm gives similarity value between any input image pair. The detailed explanation of matching image pair is as follows.

At first SURF features are obtained followed by matching of SURF feature points. Block numbers b1 and b2 represents where matched feature points are located in image pair. CNN features for both blocks b1 and b2 are obtained until all blocks b1 or b2 are marked. Each time the loop iterates, window size of blocks is increased and correlation between CNN features is found. Size of block depends on vale of ratio. If the ratio is less than one then block window *b1* of query image is increased by *nbsize* while block window *b2* of image from dataset is increased by factor r as shown in Eq. (2) and vice versa. Initial value of *nbsize* is same as block size. In each iteration, *nbsize* is incremented by block size as discussed in algorithm given in Fig. 3.

$$\begin{aligned} r &= round(1/ratio) \times nbsize \;\; if\ ratio \leq 1 \\ &= round(ratio) \times nbsize \quad otherwise \end{aligned} \quad (2)$$

Each time, correlation value of CNN features of image patch window is calculated. Window is expanded till new correlation value is found to be equal or better than previous value. Same procedure is carried out for rest of the matched SURF points if no improvement is observed in correlation value. Blocks numbers for which correlation value greater than threshold are extracted by the procedure discussed in the next section. These extracted blocks are marked in order to obtain number of matched blocks.

In order to determine depth of neighbouring blocks we set *level* as input parameter to neighbour extraction algorithm. Higher the sizes of each window patch, more neighbouring blocks surrounding the current block are extracted. *level* depends on the value of r as mentioned in Eq. (2). The value of level for block b1 and b2 is calculated as mentioned Eqs. (3) and (4). Value of l is incremented by one in each loop iteration with initial value set to 1.

$$\begin{aligned} level_{b1} &= l & if\ ratio \leq 1 \\ level_{b2} &= (r/blocksize) & otherwise \end{aligned} \quad (3)$$

$$\begin{aligned} level_{b1} &= (r/blocksize) & if\ ratio > 1 \\ level_{b2} &= l & otherwise \end{aligned} \quad (4)$$

Matching process is ended if all he blocks of either image are covered from the image air.

Finally we have number of marked blocks and its corresponding correlation values for input image pair instance. These two parameters help us to calculate similarity of image pair as mentioned in Eq. (1).

3.1 Retrieval of neighbouring blocks

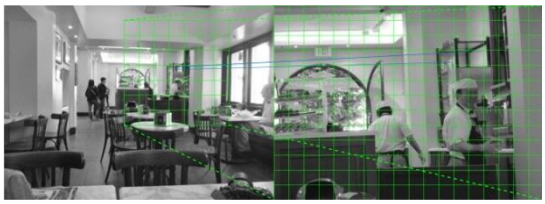
Matching process involves extracting and marking of neighbouring blocks. Our approach relies on block based matching. Entire image is divided into grid of blocks. Current block is marked

if block matching is found. Consequently we search for neighbour blocks with respect to currently matched block in order to identify near duplicate image pairs. Neighbouring blocks are extracted and marked recursively till we get match. A simple algorithm to extract neighbours is shown in Fig. 4.

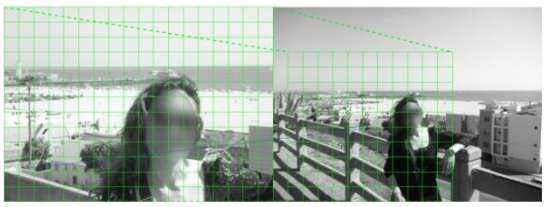
We employed recursive algorithm in order to extract neighbouring blocks. Block number and level is given as input parameter. Eqs. (3), (4), and (5) show that number of neighbouring blocks that are to be extracted by algorithm depends on the parameter r which in turn depends on value of ratio. In this way we provide flexibility while extracting different number of blocks for different zooming conditions. As stated in algorithm Nb_i contains set of all neighbouring blocks and is taken as output parameter. Initially Nb_i contains only eight surrounding neighbour blocks. More surrounding blocks are extracted and added to set Nb_i based on the value passed to input parameter *level*. Blocks that lie out of image boundary blocks are omitted by setting negative value to such blocks. Input parameter level determines how many times neighbours are extracted and added in set. Due to recursive implementation of our neighbouring algorithm, Nb_i set may result into inclusion of duplicate block numbers. To eliminate duplicate block numbers we return Nb_i with unique block numbers only.

```
Neighbours (blocknum, level)
    %input is block no. and output is neighbours of current block
        Extract surrounding 8 neighbouring blocks and assign it to Nbi
        Set Nbi to -1 for boundary cases of image
        Select non zero neighbours from Nbi
        While (level>1)
            Nbi  = Nbi || Neighbours(Nbi, 1) % add in existing set
            level  = level-1
        End
        return Nbi with unique values
```

Figure. 4 Algorithm to find neighbours for given set of blocks and level



Matching blocks for ratio <1



Matching blocks for ratio >1

Matching blocks for ratio =1

Figure. 5 Successful matching instances of extracted neighbours for different values of ratio of SURF scale

Extractions of neighbouring blocks depend on ratio of scale of SURF features of image pairs. It should be noted that extraction process depends on input parameter level which in turn proportional to ratio as mentioned earlier in matching algorithm. Final extracted and successful matching of neighbouring blocks for different cases of ratio is shown in Fig. 5. It shows improved matching compared to Fig. 7 which shows lower correlation value for similar image pairs. Fig. 5 also represents how ratio obtained from SURF scale of input image pairs are effectively utilized in order to determine neighbours of image pairs at different level of zooming. We get full coverage of blocks from image pairs despite zooming.

4. Experimental results

Our model is applied on California-ND [30] and Holiday [15] dataset. California-ND dataset contains many difficult cases of image near duplicates. Experiments are carried out in MATLAB 2017 with neural network model VGG19. System configuration is Titan XP Nvidia GPU. All the input set images were resized in multiple of block size. In our experiment 56 x 56 block size is taken for both

datasets. In case of SURF feature matching, input parameters are set to defaults in MATLAB. Sample retrieval results along with the failure case are shown in Fig 6.

By observing retrieved results of Fig. 6, it should be noted that our algorithm gives good efficiency for the various ND (near duplicate) cases such as change in viewpoint, extreme zooming condition, different background occluded objects, combination of change in viewpoint and zooming as shown in first row, second row, third row and fourth row of figure respectively. Fifth row of figure represents severe change in viewpoint leading to failure of our technique as we do not get any local matching. Fig. 7 shows some sample cases of matching CNN features of entire images in global manner. Correlation value obtained for the given sample image pairs is found to be even less than 0.5 resulted into failure of global CNN matching strategy. With our technique, it is observed that for the given image pair, all blocks are matched successfully. In order to achieve correct matching, we employed local to global matching. Fig. 5 show successful matching of image pairs giving robustness to our technique.

Figure. 6 First four rows represent sample query and corresponding top four retrieved images. First column in first four rows represents query image. In forth row, two false positives are detected. First image pair and second pair in last row show actual ND (Near Duplicate) pair and pair retrieved by our algorithm (failure case).

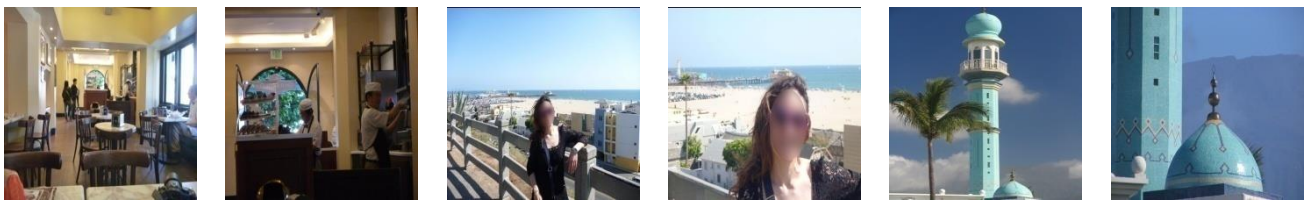


Figure. 7 Correlation values of CNN features for zoomed image pairs

We used mean average precision (*mAP*) defined in [31] and in Eq. (2) which computes the area under the precision-recall curve for each query as performance measure of our proposed approach. Mean average precision is defined as mean of average precision defined in below mentioned equation, where n is number of relevant images, r_k is rank of k-th retrieved relevant image and Q is total number of queries.

$$mAP = \frac{\frac{1}{n}\sum_{k=0}^{n} k/r_k}{Q} \quad (4)$$

California-ND is an annotated dataset containing difficult cases of near duplicate images. Dataset contains total 701 images. We considered subjects as base for measuring accuracy of our algorithm. From each subject first image is taken as query image that is compared with rest of all images. We have not taken all images as query images resulting into approximately 0.5 Million image comparison. Table 1 gives comparative analysis of mean average precision values for various cases giving remarkable performance for 10 different subjects. From each subject first image is taken as query image that is to be compared with rest of the images. We tested and compared the results of our approach for various cases 1) Incorporating SURF local feature with CNN feature and measuring similarity based on number of matched blocks found 2) Incorporating SURF and CNN features and measuring similarity based on number of matched blocks as well as correlation values of matched blocks. Map values are calculated for various subjects as shown in Table1.

Table 1. Mean average precision (*mAP*) values of different subjects of California-ND dataset

Subjects	*Mean average precision(mAP)* vales	
	CNN +SURF Block coverage based detection	Blocks coverage & sum correlation based detection
Subject0	0.6258	0.8293
Subject1	0.6346	0.8221
Subject2	0.5750	0.7242
Subject3	0.5822	0.7359
Subject4	0.6158	0.7713
Subject5	0.6255	0.8055
Subject6	0.6163	0.7762
Subject7	0.5861	0.7640
Subject8	0.5928	0.7847
Subject9	0.6385	**0.8660**
Average	0.6092	**0.7879**

According to [30], identifying near duplicate is a subjective matter. From the above table it is observed that *mAP* value varies from 0.72 to 0.86 with average value 0.78 approximately for the second case. This shows remarkable performance of our model.

Table 3 shows comparative analysis of our approach with various techniques for Holiday dataset. For Holiday dataset, we resized images to 30% with block size 56 in order to reduce comparison time. Holiday dataset contains 500 query images from the total of 1491 images. We compared our results for the Holiday dataset with other available approaches. We performed 0.74 million comparison for Holiday dataset. We compare our result with various state of art techniques. Our approach shows remarkable improvement as compared with some of the state of art techniques as mentioned in Table 3. BOW (Bag of words) [12] based methods were considered to be one of popular techniques among it. BOW based technique provides some robustness with respect to certain image transformation such as scaling, occlusion etc. However, they do not provide sufficient spatial information. This motivated us to use CNN features which maintain spatial information to better extent. As mentioned in section two, various encodings techniques such as VLAD [13], normalized VLAD (VLAD SSR- signed square root) [32] and Fisher Vectors (FVs) [14] were introduced in order to improve retrieval accuracy. Mean average precision (mAP) values shown in Table 2, reflects better performance of FV encoding compared to VLAD or VLAD SSR encoding. Encoding technique mentioned here requires use of code book while our technique does not require generation of any code book. Incorporating multiple features plays vital role in retrieval process. Fusion of VLAD vectors based on colour and texture features [33] were proposed which gives same performance as ours in which combined use of SURF and CNN is shown. In [34], DIFT (DCT inspired feature transform) is introduced unlike conventional local features. However VLAD retrieval model using DIFT local features gives less accuracy compared to our model. Performance of our 4096 dimensional CNN features is also comparable with Triangular embedding [35] with dimension size ranging from 128 to 8024. Local feature matching plays an important role to improve performance. SURF based colour feature (CSURF) [36] along with VLAD found to provide better results with mAP value 0.738 which shows improvement to above mentioned state of art technique. This is little less than the mAP value reported by us with SURF as local feature.

Table 2. Comparison of results with state of art techniques

Methods	Mean Average Precision (mAP)
BOW based [12]	0.597
VLAD based [13]	0.510
Ours(Block coverage based)	0.551
VLAD+SSR [32]	0.557
Fisher based [14]	0.565
CNN Based [38]	0.612
DIFT based (DCT Inspired Feature Transform) [34]	0.7
Instance retrieval with deep convolutional networks[37]	0.716
CSURF+VLAD [36]	0.738
Neural code based layer 'fc7' [21]	0.7-0.76
Ours(Blocks coverage & sum correlation based detection)	0.744
Fusion of VLAD vectors based on colour and texture [33]	0.7499
Triangular embedding D=8064[35]	0.771
Triangular embedding D=128[35]	0.617
Multi-scale order less pooling dimension =512 [22]	0.783

In following discussion, comparison is made with techniques where deep features are utilized for retrieval. Razavian [37] suggested cross matching strategy of each regions incurring requirement of high memory and time. Our model being adaptive extracts single CNN vector for given instance of time as well as having better performance than the model proposed in [37]. In [21], author adapted retraining in order to achieve better retrieval performance. Our model does not undergo any sort of retraining and simply used CNN features trained on Imagenet giving comparable performance. Due to fine tuning, their performance [21] is observed little more than ours. In [22], CNN features of patches at different levels are extracted followed by applying VLAD at each level. This results into very high dimensional feature vector (12,288) with *mAP* value 0.78 better than *mAP* reported by us. However here, it should be noted that our retrieval results can be considered nearly same as reported in [22] for 4096-dimensional feature.

Like our approach, raw image pairs are processed directly and similarity is found out using adopted neural network model in [38]. Like our model, the model discussed in [38], does not incorporate patch level or object level matching. As a result, the reported mAP value is 0.612 which is found to be 21% less compared to our model.

5. Conclusion

In this paper, we presented a technique for matching and retrieving near duplicate images in adaptive manner. Local patch matching process using the power of CNN handles various difficult cases of near duplicate images. Matching global CNN features for the entire image may fail for various cases of near duplicate retrieval. We utilized the power of local match and moved towards global match to eliminate the drawbacks of algorithms based on only local features as well as based on only global features. Our model is able to achieve significant matching of image patches at considerable level of zooming for the given image pairs in order to retrieve near duplicate images. Our adaptive model does not need any sort of training and is applicable to any category of images as well as it does not require storage of any pre-computed image descriptors. We found the model to be better at retrieval with mean average precision value 0.74 which is better than the value reported in recent paper [32] for the same Holiday dataset. For California-ND dataset we get mean average precision value up to 0.86 which is remarkable.

Limitation

Our approach relies on matching local features and then subsequently CNN features of surrounding regions are compared. If matching of local features does not succeed then CNN feature matching is not carried out. Images with significant changes in viewpoint may lead to failure of local feature matching. Affine invariant local features may be tried out to obtain better results. Other limitation of our model is comparison time is little more due to adaptive nature of our model. Processing may be done on cloud in order to achieve faster results.

Acknowledgments

We gratefully acknowledge CHARUSAT University and the support of NVIDIA Corporation with the donation of Titan Xp GPU used for this research.

References

[1] Y. Hu, X. Cheng, L. Chia, X. Xie, D. Rajan, and A. Tan, "Coherent Phrase Model for Efficient Image Near-Duplicate Retrieval", *IEEE Transaction on Multimedia*, Vol.11, No.8, 2009.

[2] X. Wang, L. Zhang, and C. Liu, "Duplicate Discovery on 2 Billion Internet Images", In: *Proc. of IEEE Conference on Computer Vision*

and Pattern Recognition Workshops, pp.429-436, 2013.

[3] L. A. Khan and M. S. Ahmed, "A Novel Technique Using Multiple K-Shingling Based Weighted Dissimilarity Score for Web Content Outlier Mining", *International Journal of Intelligent Engineering and Systems*, Vol.12, No.4, pp.244-254, 2019.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks", *Advances in neural information processing systems*, Vol. 60, No. 4, pp.1097-1105, 2012.

[5] H. Bay, A. Ess, T. Tuytelaars, and L. G. Van, "Speeded up robust features (SURF)", *Computer Vision and Image Understanding*, Vol. 110, No. 3, pp.346-359, 2008.

[6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", In: *Proc. of International Conference on Learning Representations*, 2015.

[7] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", *International Journal of Computer Vision*, Vol.60, No.2, pp.91-110, 2004.

[8] Y. Ke, R. Sukthankar, and L. Huston, "An efficient parts-based near-duplicate and sub-image retrieval system", In: *Proc. of the 12th annual ACM international conference on Multimedia*, pp. 869-876, 2004.

[9] Y. Ke and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors", In: *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.506-513, 2004.

[10] J. Foo and R. Sinha, "Pruning SIFT for scalable near-duplicate image matching", In: *Proc. of the Eighteenth Conference on Australasian Database*, Vol. 63, pp.63-71. 2007.

[11] V. Pimenov, "Fast Image Matching with Visual Attention and SURF Descriptors", In: *Proc. of the 19th International Conference on Computer Graphics and Vision*, pp. 49-56, 2009.

[12] D. Nister and H. Stewenius, "Scalable Recognition with a Vocabulary Tree", In: *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.2161-2168, 2006.

[13] H. Jégou, M. Douze, C. Schmid, and P. Perez, "Aggregating local descriptors into a compact image representation", In: *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.3304-3311, 2010.

[14] F. Perronnin, Y. LIU, J. Sánchez, and H. Poirier, "Large-scale image retrieval with compressed Fisher vectors", In: *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.3384-3391, 2010.

[15] H. Jegou, M. Douze, and C. Schmid C, "Hamming Embedding and Weak Geometric Consistency for Large Scale Image Search", In: *Proc. of European Conference on Computer Vision*, pp.304-317, 2008.

[16] Y.G. Jiang and C.W. Ngo, "Visual word proximity and linguistics for semantic video indexing and near-duplicate retrieval", *Computer Vision and Image Understanding*, Vol.113, No.3, pp.405-414, 2009.

[17] W.L. Zhao and C.W. Ngo, "Scale-rotation invariant pattern entropy for keypoint-based near-duplicate detection", *IEEE Transaction on Image Processing*, Vol.18, No.2, pp.412-423, 2009.

[18] E. Younessian, D. Rajan, and E. S. Chng, "Improved Keypoint Matching Method for Near-Duplicate Keyframe Retrieval", In: *Proc. of the 11th IEEE International Symposium on Multimedia*, pp. 298-303, 2009.

[19] D. Zhang and S. Chang, "Detecting image near-duplicate by stochastic attributed relational graph matching with learning", In: *Proc. of the 12th annual ACM international conference on Multimedia*, pp.877-884, 2004.

[20] Y. Li, X. Kong, L. Zheng, and Q. Tian, "Exploiting Hierarchical Activations of Neural Network for Image Retrieval", In: *Proc. of the 24th ACM international conference on Multimedia*, pp.132-136, 2016.

[21] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky, "Neural Codes for Image Retrieval", In: *Proc. of the 13th European Conference on Computer Vision, Zurich, Switzerland, Lecture Notes in Computer Science, Springer*, Vol. 8689, pp.584-599, 2014.

[22] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features", C*omputing Research Repository*, Vol. abs/1403.1840, 2014.

[23] K. R. Mopuri and R. V. Babu, "Object level deep feature pooling for compact image representation", In: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp.62-70, 2015.

[24] L. Xie, R. Hong, B. Zhang, and Q. Tian, "Image Classification and Retrieval are ONE", In: *Proc. of the 5th ACM on International Conference on Multimedia Retrieval*, pp.3-10, 2015.

[25] J. Revaud, W. Philippe, Z. Harchaoui, and C. Schmid, "Deepmatching: Hierarchical deformable dense matching", *International Journal of Computer Vision*, Vol.120, No.3, pp. 300-323, 2016.

[26] K. Yan, Y. Wang, D. Liang, T. Huang, and Y. Tian, "CNN vs. SIFT for Image Retrieval: Alternative or Complementary?", In: *Proc. of the 24th ACM international conference on Multimedia*, pp.407-411, 2016.

[27] C. L. Zitnick and P. Dollár, "Edge Boxes: Locating Object Proposals from Edges", In: *Proc. of the 13th European Conference on Computer Vision*, pp.391-405, 2014.

[28] V. Chandrasekhar, J. Lin, O. Morère, H. Goh, and A. Veillard, "A practical guide to CNNs and Fisher Vectors for image instance retrieval", *Signal Processing*, Vol.128, No.C, pp.426-439, 2016.

[29] L. Zheng, S. Wang, J. Wang, and Q. Tian, "Accurate Image Search with Multi-Scale Contextual Evidences", *International Journal of Computer Vision*, Vol.120, No.1, pp.1-13, 2016.

[30] A. Jinda-Apiraksa, V. Vonikakis, and S. Winkler, "California-nd: An annotated dataset for near duplicate detection in personal photo collections", In: *Proc. of the Fifth International Workshop on Quality of Multimedia Experience*, pp.142-147, 2013.

[31] X. Wu, A. G. Hauptmann, and C. W. Ngo, "Practical elimination of near-duplicates from web video search", In: *Proc. of the 15th ACM international conference on Multimedia*, pp.218-227, 2007.

[32] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid, "Aggregating Local Image Descriptors into Compact Codes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.34, No.9, pp.1704-1716, 2012.

[33] Y. Wang, Y. Cen, R. Zhao, S. Kan, and S. Hu, "Fusion of multiple VLAD vectors based on different features for image retrieval", In: *Proc. of IEEE 13th International Conference on Signal Processing (ICSP)*, pp. 742-746, 2016.

[34] Y. Wang, M. Shi, S. You, and C. Xu, "DCT Inspired Feature Transform for Image Retrieval and Reconstruction", *IEEE Transactions on Image Processing*, Vol.25, No.9, pp.4406-4420, 2016.

[35] H. Jégou and A. Zisserman, "Triangulation Embedding and Democratic Aggregation for Image Search", In: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp.3310-3317, 2014.

[36] E. Spyromitros-Xioufis, S. Papadopoulos, I. Y. Kompatsiaris, G. Tsoumakas, and I. Vlahavas, "A comprehensive study over vlad and product quantization in large-scale image retrieval", *IEEE Transactions on Multimedia*, Vol.16, No.6, pp.1713-1728, 2014.

[37] A. S. Razavian, J. Sullivan, S. Carlsson, and A. Maki, "Visual Instance Retrieval with Deep Convolutional Networks", *ITE Transactions on Media Technology and Applications*, Vol.4, No.3, pp.251-258, 2016.

[38] Y. Zhang, Y. Zhang, J. Sun, H. Li, and Y. Zhu, "Learning Near Duplicate Image Pairs using Convolution Neural Networks", *International Journal of Performability Engineering*, Vol.14, No.1, pp.168-177, 2018.