International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:5, No:2, 2011

Sequential Straightforward Clustering for Local Image Block Matching

Mohammad Akbarpour Sekeh ,Mohd. Aizaini Maarof, Mohd. Foad Rohani, Malihe Motiei

Abstract—Duplicated region detection is a technical method to expose copy-paste forgeries on digital images. Copy-paste is one of the common types of forgeries to clone portion of an image in order to conceal or duplicate special object. In this type of forgery detection, extracting robust block feature and also high time complexity of matching step are two main open problems. This paper concentrates on computational time and proposes a local block matching algorithm based on block clustering to enhance time complexity. Time complexity of the proposed algorithm is formulated and effects of two parameter, block size and number of cluster, on efficiency of this algorithm are considered. The experimental results and mathematical analysis demonstrate this algorithm is more cost-effective than lexicographically algorithms in time complexity issue when the image is complex.

Keywords—Copy-paste forgery detection, Duplicated region, Time complexity, Local block matching, Sequential block clustering.

I. Introduction

WITH today's advance and widely accessible image editing software, it is so easy to manipulate digital image. Unfortunately, it often causes to creation of forged images. This can reduce the reliability of the digital image. As digital images are one of the most important and useful information in some area such as forensic investigation, criminal investigation, insurance services, surveillance systems, intelligence services and other information system organizations, image forgery detection was created. Digital image forgery detection system is to discover evidence of tampering by scrutinizing the forgery's clues on the image.

As several existing types of forgery on the image, there are several methods to explore the digital image forgeries. Based on literature, it can be found that researchers are trying to detect the image forgeries via following proposed techniques: Duplicated region detection [1]- [2]- [3], Noise inconsistency [4]- [5], Light inconsistency [6], Color filter array processing [7] and Traces of the re-sampling detection [8].

In this paper we scrutinize copy-paste forgery's clues and focus on duplication region detection. Duplicated region detection has two main facing open problems: The first problem is robustness and accuracy of the detection against under-modification operations such as rotation, noising, compression and retouching. Another important problem in duplicated region detection is high time complexity of the block matching step.

This paper concentrates on improving time complexity of the forgery detection with proposing a local block matching technique and is organized in five major parts. The first part is begun with introducing the image forgery detection and scope of the study. In part II, related works, duplicated region detection, research area, problems and block clustering techniques will be overviewed. Part III explains the local block matching algorithm based on sequential block clustering in order to reduce computational time. In part IV, we formulate the time complexity function of proposed algorithm and mathematically analysis for comparing the algorithm with previous methods. And finally, we bring conclusion of the research and explain the future works.

II. RELATED RESEARCH

The first publication in copy-move forgery detection area has been proposed by Fridrich in [1]. The paper proposed DCT coefficients as block presentation method and detection of the duplicated region was based on matching the quantized lexicographically sorted discrete cosine transform coefficients of blocks. The lexicographically sorting has complexity in order of $O(MNlogMN)$ that M and N are width and height of image.

The next two papers, Popescu and Farid [2] proposed Principal Component Analysis (PCA) instead of DCT to reduce the block feature dimension and Bayram [9] proposed Fourier Mellin Transform (FMT) to enhance the robustness against scaling and rotation, also applied lexicographically sorting for finding the duplicated regions in the image.

Bayram [9] also proposed a new method to improve the efficiency and reduce the time complexity namely Counting Bloom Filters with hashing the feature vectors. However, finding the effective hash function is not easy and also this technique affects robustness of the system.

Babak Mahdian [10] proposed blurring invariant feature to enhance the robustness against blurring and kd-tree presentation for improving the complexity of the matching step. As complexity of kd-tree depends on the distribution of similar intensity blocks, one of the important problems in Babak's method is yet high computational time [11]- [12].

Hwei-jen Lin et.al [13] proposed a new block feature extraction method. They represented each overlapping blocks by 9-dimensional feature vector in spatial domain for improving the robustness against compression and noising. They also applied efficient Radix-sort for performing the lexicographically sorting with order of O(nk). The radix-sort limits type

Mohammad Akbarpour Sekeh is working on image forgery detection area, Email: asmohammad4@live.utm.my.

Mohd. Aizaini Bin Maarof is focused Network and Multimedia Forensic, Email: aizaini@utm.my

Mohd. Foad Rohani is working on Intrusion Detection Systems, Email: foad@utm.my

Malihe Motiei is doing Information Security Awareness measuring, Email: mmalihe2@live.utm.my

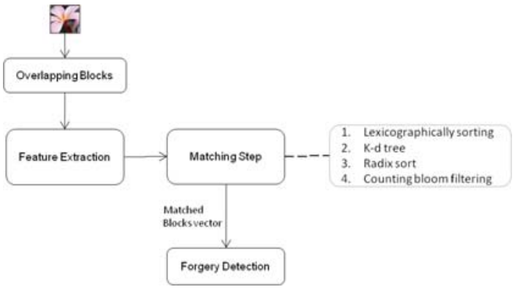


Fig. 1. Common Framework for Copy-paste Image Forgery Detection

of feature vector elements to integer and can't be always used with other different feature vectors.

As we can see in the literature review, one of the main open problems is computational time for detecting the similar blocks in matching step. Hence sequential block clustering for local block matching will be proposed to improve time complexity of block matching. For this purpose, in the next two sections, duplicated region detection and block clustering techniques are overviewed.

A. Duplicated Region Detection

Duplicated region detections as one of the forgery detection techniques signify copy move (copy-paste) forgery on the image. This type of forgery is to clone portion of an image to conceal or duplicate specific objects. The copy-paste forgery brings into the image several near-duplicated image regions. It is important to note that duplicated regions mostly are not exactly alike, because skilled forger usually modifies the copied regions by some extra editing operations such as rotation, noising, compression and blurring. Common framework of this type of forgery detection has been proposed by Fridrich [1]. Fig. 1 shows this framework that has been followed by other researchers. As shown in Fig. 1, we can classify the framework components in 4 major steps: overlapping blocks, feature extraction, matching step and forgery detection.

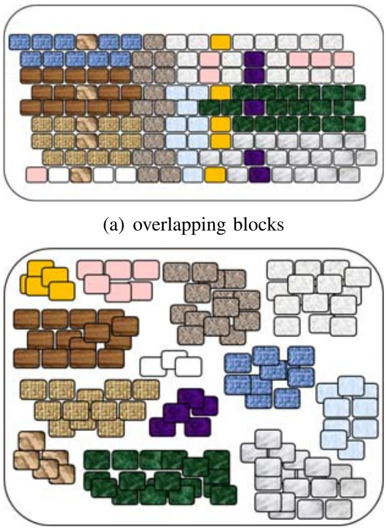
The first step is overlapping blocks wherein the image is divided to several overlapping blocks in size of Blocksize bb. so, the image with $M \times N$ pixel and Blocksize b can be divided to $(M-b+1) \times (N-b+1)$ overlapping blocks. In the second step of forgery detection, feature extraction, best and robust feature of block should be extracted. The result of this step can be a matrix wherein each row includes the block feature vector. In block matching step, all elements in feature vector matrix should be sorted to find every similar blocks. Since block similarity detection in huge number of blocks has high computational time, improving time complexity is an open problem in this type of forgery detection. The last step is forgery detection wherein matched blocks are reanalyzed to find the exact forgeries. This step is needed, because all the matched blocks don't signify forgery on the image.

B. Sequential Block Clustering

Block clustering is grouping of similar blocks. The blocks of one cluster should be similar to one other and dissimilar to the blocks of other clusters as shown in Fig. 2. Clustering algorithms may be divided into the following major categories:

1) Sequential algorithms
2) Hierarchical algorithms
3) Algorithms based on cost-function optimization

Fastest method of clustering is straightforward sequential algorithm that produces a single clustering.



(a) overlapping blocks

(b) Clustered Blocks

Fig. 2. Example of Block Clustering

In this type of clustering all the feature vectors are presented to the algorithm once or a few times. In this case number of clusters is not known at the first. In fact new clusters are created as the algorithm evolves. The user-defined parameter required by the algorithm is the threshold of dissimilarity of new item and cluster feature [14].

Block classification differs from region classification and image segmentation. Region classification algorithm such as k-mean clustering is restricted to data for which there is a notion of a center. In the block classification, each block is presented to the clustering algorithm in order to find the first similar group. If there is no similar cluster with new block, algorithm creates a new cluster and puts the block in it.

III. LOCAL BLOCK MATCHING

As mentioned in Part II section A, in matching step, block feature vectors should be analyzed to find similar blocks. In the most of the existing publications exhaustive search or lexicographically sorting has been used to detect the similar blocks in matching step. However, each block should be compared with all other blocks in feature vector matrix. Therefore one important problem in block matching is number of extra and ineffective matching that extremely affects on growing computation time. The question is how to reduce these extra matching operations without effect on accuracy. We proposed a local block matching algorithm to overcome this problem. In order to effectively decreasing the number of extra matching, we perform the local block matching based on block clustering. Therefore, after extracting the low and high accurate features of block, we use a sequential clustering algorithm to group all overlapping blocks. The proposed workflow is shown in Fig. 3.

In our proposed method, two types of feature extraction method are applied. The first one is low accurate feature with low feature vector dimension for block clustering and the next feature is robust and high accurate feature for exact local block matching. However, in this paper we look at both features as black box. In the next section local block matching algorithm is formulated.

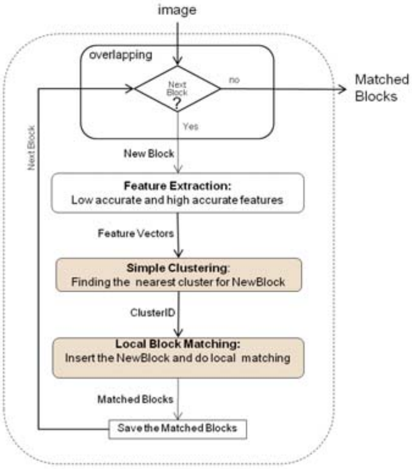


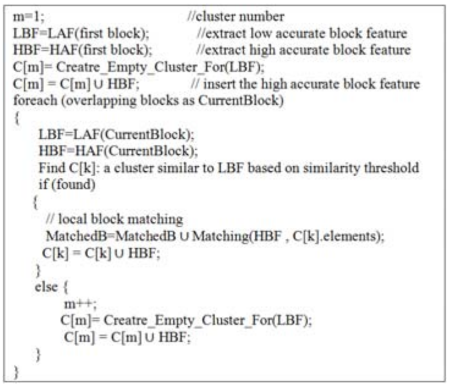
Fig. 3. Workflow of proposed Local Block Matching

A. Algorithm Formulation

As shown in Fig. 3, the output of local block matching is a matched block vector. The local block matching algorithm can be formulated as follow:

1) Divide Image into overlapping blocks
2) Extract the low accurate feature and high accurate feature of first block as LBF and HBF
3) Create empty BST structure C for clusters
4) Create new cluster c similar to LBF in C.
5) Create empty BST structure CB for elements of cluster (each cluster may have several blocks)
6) Insert the first block feature HBF in c.
7) For each overlapping block as CurrentBlock do the following step.
8) Extract the LBF and HBF of Current block.
9) Find a cluster similar to LBF based on sequential block clustering algorithm.
10) If c_k was found do exact block matching between HBF of CurrentBlock and cluster's blocks.
11) The matched block will save in the Matchedblock vector.
12) Insert the block feature HBF in c_k.
13) If c_k wasn't found create empty cluster c_k and insert the block feature HBF in it.

The result of this algorithm is Matchedblock vector to identify the similar blocks. This vector should be present into forgery detection algorithm to find the exact forged blocks. The following pseudocode explains components of Fig. 3 in details:



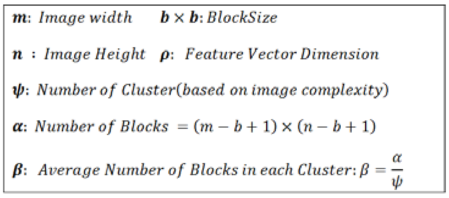
```
m=1;                          //cluster number
LBF=LAF(first block);         //extract low accurate block feature
HBF=HAF(first block);         //extract high accurate block feature
C[m]= Creatre_Empty_Cluster_For(LBF);
C[m] = C[m] ∪ HBF;            // insert the high accurate block feature
foreach (overlapping blocks as CurrentBlock)
{
    LBF=LAF(CurrentBlock);
    HBF=HAF(CurrentBlock);
    Find C[k]: a cluster similar to LBF based on similarity threshold
    if (found)
   {
      // local block matching
      MatchedB=MatchedB ∪ Matching(HBF , C[k].elements);
     C[k] = C[k] ∪ HBF;
   }
   else {
         m++;
        C[m]= Creatre_Empty_Cluster_For(LBF);
         C[m] = C[m] ∪ HBF;
   }
}
```

IV. ALGORITHM ANALYSIS AND RESULT

In order to evaluate our algorithm, we compute the time complexity function of proposed algorithm to find the exact order of complexity. We compare this algorithm with previous works by time complexity function.

A. Time Complexity of Proposed algorithm

We define following variables as initial declaration to compute time complexity function:



$\boldsymbol{m}$: *Image width* $\boldsymbol{b} \times \boldsymbol{b}$: *BlockSize*

$\boldsymbol{n}$: *Image Height* $\boldsymbol{\rho}$: *Feature Vector Dimension*

$\boldsymbol{\psi}$: *Number of Cluster(based on image complexity)*

$\boldsymbol{\alpha}$: *Number of Blocks* $= (m - b + 1) \times (n - b + 1)$

$\boldsymbol{\beta}$: *Average Number of Blocks in each Cluster*: $\beta = \frac{\alpha}{\psi}$

Based on the proposed algorithm, complexity of each part is as follow:

1. Dividing the image into overlapping blocks is $O(\alpha)$

2. Block feature extraction is $O(\alpha b^2)$

3. Complexity of sequential block clustering with α blocks utilizing BST structure is formulated as follow:

Note that complexity of search operation in BST is in order of $O(log_2 n)$.

$$T_1(\alpha, \psi) = log_2(1) + log_2(2) + log_2(3) + \ldots + log_2(\psi) + log_2(\psi) + \ldots + log_2(\psi)$$

So after summation and simplification with Stirling's approximation [15]:

$$T_1(\alpha, \psi) = log_2(\psi!) + (\alpha - \psi)log_2(\psi) \in O(\alpha log_2(\psi)) \quad (1)$$

4. Complexity of local block matching with BST structure is as follow:

$$T_2(\alpha, \psi, \rho) = \rho\psi(log_2(1) + log_2(2) + \ldots + log_2(\beta)) = \rho\psi log_2(\beta!)$$

$$T_2(\alpha,\psi,\rho) = \rho\alpha\frac{log_2(\beta!)}{\beta} \qquad (2)$$

The total time complexity of the system is in order of:

$$T(\alpha,\psi,\rho) \in O(\alpha + \alpha b^2 + \alpha log_2(\psi) + \rho\alpha\frac{log_2(\beta!)}{\beta}) \qquad (3)$$

For simplifying the equation we use Stirling's approximation:

$$n! \cong ((\frac{n}{e})^n\sqrt{2\pi n} \qquad (4)$$

We take logarithm in both side of (4) and then divide by n:

$$log_2(n!) \cong nlog_2(n) - n + \frac{1}{2}log_2(n) + C$$

$$\frac{log_2(n!)}{n} \in O(log_2(n)) \qquad (5)$$

So the matching algorithm has complexity in order of $O(\alpha log_2(\psi)) + \rho\alpha log_2(\beta)$ and total time complexity of proposed method after simplifying is in order of:

$$T(\alpha,\psi,\rho) \in O(\alpha + \alpha b^2 + \alpha log_2(\psi) + \rho\alpha log_2(\beta)) \qquad (6)$$

B. Discussion and Evaluation

Time complexity of algorithms is applied to analysis and algorithm evaluation. In the previous section, the time complexity function of proposed algorithms was computed. Refer to (6), computational time is function of number of block α, feature vector dimension ρ and number of cluster ψ. Here we divide the complexity function to three cases based on ψ :

$$T(\alpha,\psi,\rho) \in \begin{cases} O(\alpha\log_2(\alpha) & \psi = \alpha, \\ O(\alpha\log_2(\psi) + \rho\alpha\log_2(\beta)) & 1 < \psi < \alpha, \\ O(\rho\alpha\log_2(\alpha)) & \psi = 1. \end{cases} \qquad (7)$$

The first case is $\psi = 1$ wherein the image can't be clustered. This is the worst case of our algorithm. The complexity of this case is in order of complexity in lexicographically sorting. The next case is $\psi = \alpha$ wherein the number of cluster is maximum. This is the best case of proposed algorithm wherein it is no need to use the high accurate feature vector.

Here, time complexity of proposed algorithm is compared with time complexity of lexicographically sorting as shown in Table 1. Fridrich in [1] applied the lexicographically sorting and several researchers also used it for their matching steps.

TABLE I
TIME COMPLEXITY

Paper	Time complexity	Matching Method
Common Framework	$O(\rho\alpha\log_2(\alpha) + \rho_2\alpha)$	Lexicographically sorting
Proposed Framework	$O(\alpha\log_2(\psi) + \rho\alpha\log_2(\beta))$	Local Block Matching

Fig. 4 shows the complexity growing charts based of block size wherein number of cluster ψ is assumed to be 50 and fixed. The result demonstrates that proposed method is more efficient than lexicographically sorting approach.

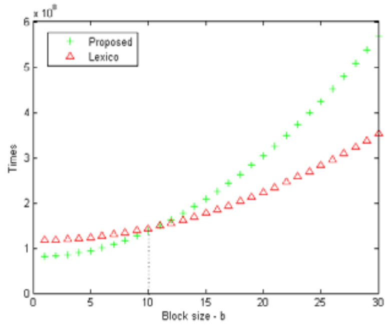


Fig. 4. Growth chart of time complexity based on block size

Fig. 5 shows the complexities functions based on number of cluster wherein number of block is constant variable. As the Fig. 5 shown, time complexity of our algorithm is dropped as the number of cluster increased.

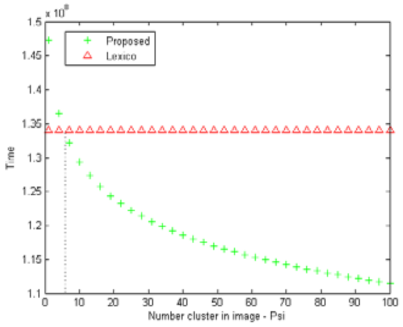


Fig. 5. Growth chart of time complexity based on number of clusters

V. Conclusion

One of the major problems in the copy-paste image forgery detection system is computational time of block matching to find similar blocks. In this paper a cost effective local block matching was proposed. This algorithm needs two types of block feature: low accurate feature and high accurate feature. Low accurate feature of blocks is one by one presented into the clustering algorithm to find the best similar cluster. The high accurate feature is presented to the matching algorithm. Because the exact block matching is locally performed, it causes to reduce unnecessary block comparing operations. This can significantly improve time complexity. The experimental results and mathematically analysis demonstrate that time complexity of proposed method is more efficient than other methods when the image is more complex. In the future we are going to consider multilayer feature extraction and also analyze effects of feature vector dimension on efficiency of local block matching.

Acknowledgment

This research is supported by Universiti Teknologi Malaysia (UTM) under Fundamental Research Grant Scheme (FRGS) with vote number 78632.

References

[1] J. Fridrich, "Detection of copy-move forgery in digital images," in *Proceedings of Digital Forensic Research Workshop*, 2003, pp. –.

[2] Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image regions," Tech. Rep., 2004.

[3] W. Luo, "Robust detection of region-duplication forgery in digital image," in *IEEE Computer Society Washington, DC, USA*, vol. Proceedings of the 18th International Conference on Pattern Recognition - Volume 04, 2006, pp. 746 – 749–.

[4] Lukas, Fridrich, and Goljan, "Detecting digital image forgeries using sensor pattern noise," in *Security, Steganography, and Watermarking of Multimedia Contents VIII*, vol. 6072, San Jose, CA, 2006, pp. –.

[5] B. Mahdian, "Using noise inconsistencies for blind image forensics," vol. 27, pp. 1497–1503–, 2009.

[6] M. K. Johnson and H. Farid, "Exposing digital forgeries in complex lighting environments," vol. 2, pp. 450–461–, 2007.

[7] Popescu and H. Farid, "Exposing digital forgeries in color filter array interpolated images," vol. 53, pp. 3948–3959–, 2005.

[8] A. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of re-sampling," Tech. Rep., 2003.

[9] S. Bayram, "An efficient and robust method for detecting copy-move forgery," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. Washington, DC, USA, 2009, pp. –.

[10] B. Mahdian and S. Saic, "Detection of copy-move forgery using a method based on blur moment invariants," vol. 171, pp. 180–189–, 2007.

[11] Z. Zhang, "A survey on passive-blind image forgery by doctor method detection," in *Proceedings of the Seventh International Conference on Machine Learning and Cybernetics*. Kunming: IEEE, 2008, pp. –.

[12] B. Mahdian, "A bibliography on blind methods for identifying image forgery," vol. Signal Processing: Image Communication, pp. 389399–, 2010.

[13] H. J. Lin, C. W. Wang, and Y. T. Kao, "Fast copy-move forgery detection," vol. 5, pp. 188–197–, 2009.

[14] T. Sergois and Koutroumbas, *Pattern Recognition - book - Third Edition*. San diego: Elsevier, 2006.

[15] Stirling, "Stirling approximation for n!" pp. –.