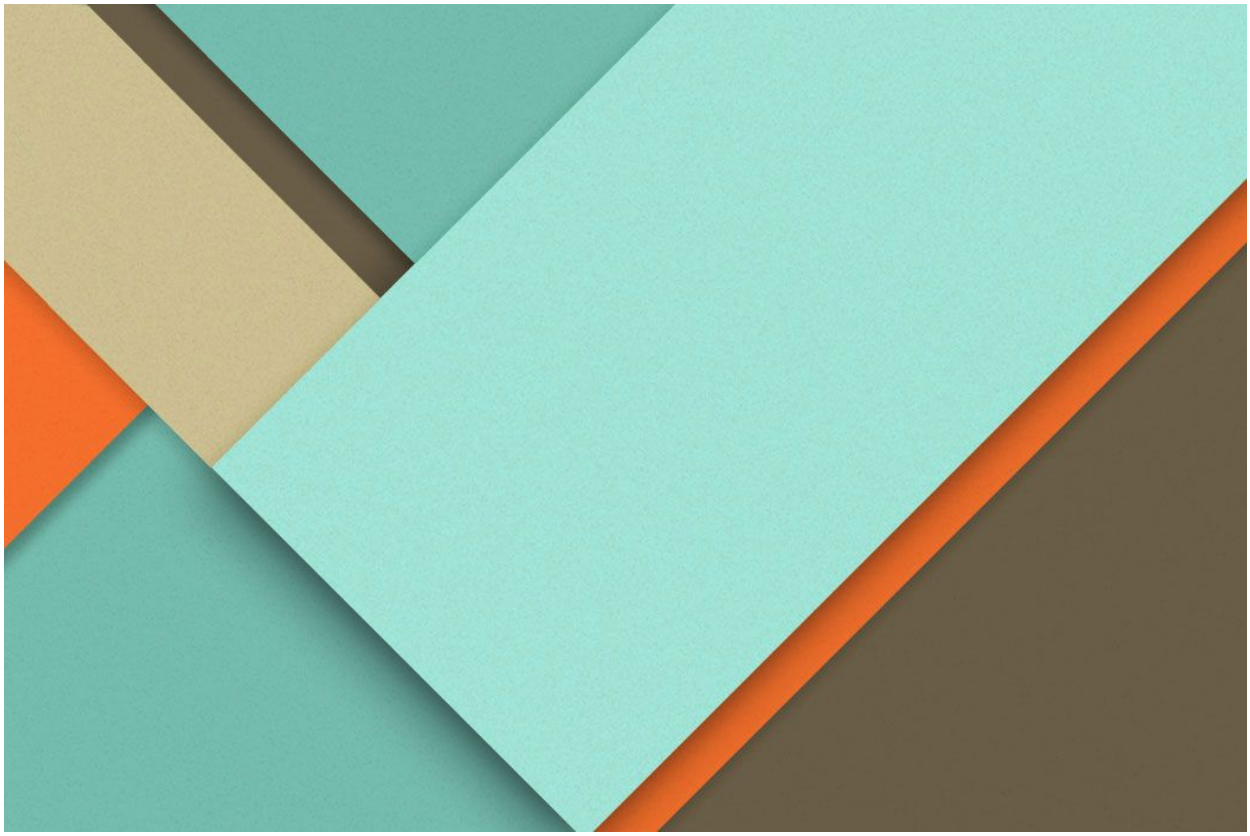

Automatic Signature Stability Analysis And Verification Using Local Features

27.09.2019

Teja Sai Dhondu(20171075)

Swetanjal Dutta(20171077)

Nishant Sharma(20171079)

Project ID: #48

Github Link:

https://github.com/swetanjal/Automatic-Signature-Stability-Analysis-And-Verification-Using-Local-Features

Main Goals of the Project

1. To have a fully automatic system that can classify signatures as genuine, forged, disguised provided a few reference signatures are available in database.
2. Based on the implementation of the following paper: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6981088&tag=1
3. Achieve an error rate of <= 15% on the 4NSigComp2010 dataset, the most well known publicly available dataset of forensic signature verification competition

Problem Definition

Present a novel signature stability analysis based on signature's local / part-based features that can classify not only between genuine and forged signatures but also can recognise disguised signature. The signatures being classified must have their ground truth signature(reference signature) in our database.

We say a signature is disguised if someone signs with the intention of hiding his/her identity later.

Our project will be tested on the famous 4NSigComp2010 dataset. This is the first ever publicly available dataset containing disguised signatures.

We are yet to choose the implementation language.

Algorithm

1. One of the unique features of the algorithm presented in this paper is that it makes use of local features of signature to classify it as genuine, fraud or disguised instead of globally classifying it.
2. When someone does a forged signature, it is observed that although there maybe global similarity with the reference signature, the local features are extremely

different as shown below by the heat maps

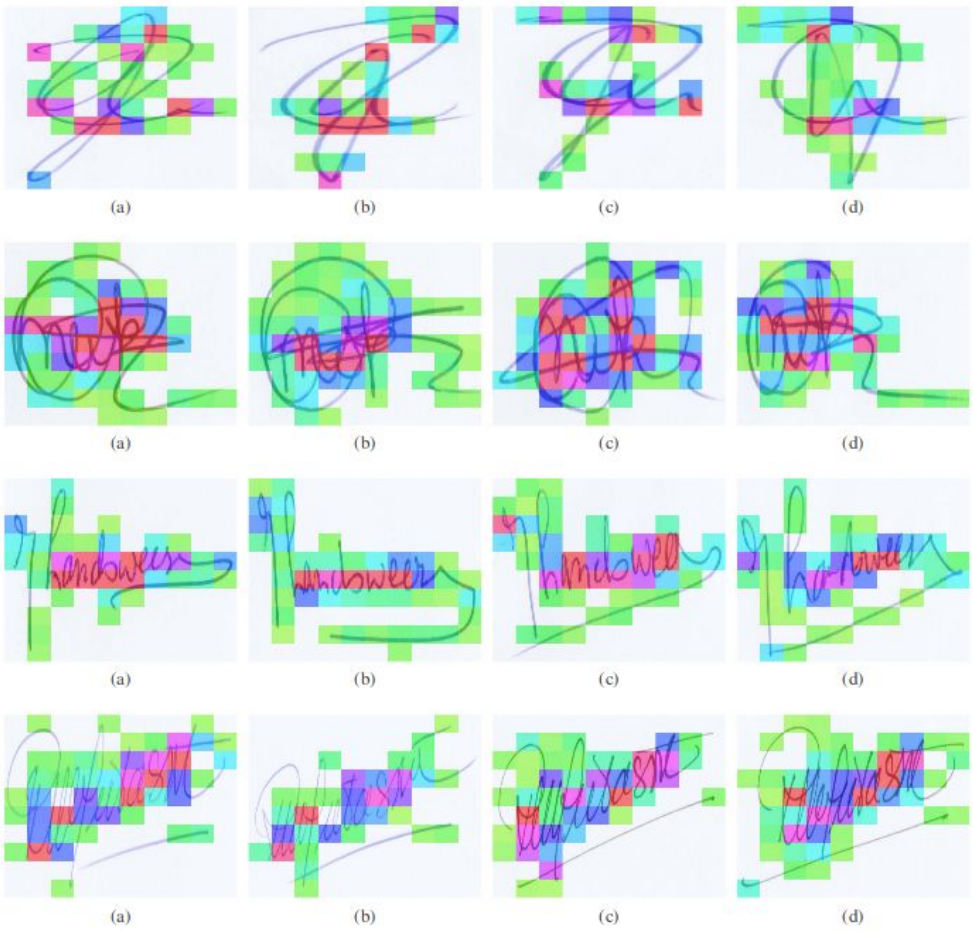

Figure 1. Heat maps of some example specimen (genuine) signatures from four different authors (one genuine author in each row) showing the most stable (green) and the most unstable (red) parts along with the moderately stable parts (colors varying from green to red through blue).

3. The local features are extracted using the famous SURF algorithm(very similar to SIFT)
4. Compute the key points from all the genuine specimen signatures of an author except only one genuine reference signature and make a temporary key points database.
5. Compute the key points from the remaining genuine reference signature and compare the distances of all of its key points from the temporary key points database.
6. Find the average distance and then mark all key points having distance less than or equal to the average distance as green and all other keypoints red.
   The below image shows a visual representation of this step.

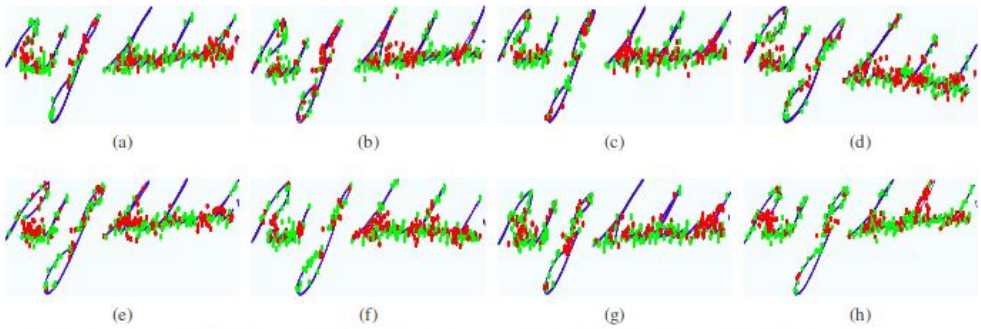

Figure 2. Example genuine reference signatures of one author. Green points are considered to be stable and are added to the reference keypoints database for performing verification. Red points are considered unstable and are not included in the reference keypoints database.

7. Repeat the above step for every genuine specimen signature in an 'n-1 cross validation manner' and populate a final reference key points database using only the stable key points(corresponding to the green regions in the visual example above) from different reference/specimen signatures.
8. Once the final reference key points database is created, key points and descriptors are extracted for the query/questioned signature. A comparison is made between the query signature key points and the key points present in our final reference key points database for that particular author.
9. Find the local key points from the query signature by using SURF. Then take the first keypoint of the query Image and compare it with all the features present in the final reference key points database, one by one. If a query signature keypoint is at a distance less than an empirically found threshold θ, note the keypoint as a match. Keep this process going until all the query signature's keypoints are traversed.
10. Finally, calculate the probability of each query signature being genuine by considering the total number of query key points and the query key points matched with the final reference key points database. This represents the average local features of the questioned signature that are present in final reference key points database of that author.

Results of the Project

1. Have a full fledged automatic system classifying between genuine, forged, disguised signatures which can be used for fraud detection in various areas.
2. Reference signatures can be added to the database with ease enabling use of our system in industry.
3. Achieve an error rate of <= 15% on the 4NSigComp2010 competition dataset.

MILESTONE	DUE DATE TIMELINE
Choose an implementation language	1st October, 2019
Learn about SURF by	5th October, 2019
Rough Implementation	10th October, 2019
Testing Phase I	12th October, 2019
Final Implementation	22nd October, 2019
Testing Phase II	24th October, 2019
Fix bugs found in previous testing phases	26th October, 2019
Final Testing	28th October, 2019
Documentation	28th October, 2019
Final commit	TBD depending on Project Submission deadline