**Washington University in st.louis**

**CSE 559a: Computer Vision**

**Final project**

**Face Alignment**

**Ding Yiheng & Wan Ren**

**Introduction:**

This project is based on “Face alignment at 3000 FPS via regressing local binary features” by Shaoqing Ren, CVPR 2014.

This paper present a highly efficient and accurate approach for face alignment by using three major techniques, random forest , local binary feature and global regression. And it achieves a very high FPS(frame per second) even in a mobile device for face alignment.

The major challenges solved by this paper are :**1) Accuracy**, robust for complex and varied face, through training a various and balanced dataset and derive specific random forest for each landmark , it also relies on the scale and rotation – invariant feature, the local binary feature. **2)speed,** on some specific dataset and resolution of images tested, the FPS can reach 3000 or even more. More than that, this method can achieve 300+ FPS on mobile device. The extremely high efficiency based on the sparse random forest and limited regression times.

**Related work and Reference:**

[1] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar.Localizing parts of faces using a consensus of exemplars.In *Computer Vision and Pattern Recognition (CVPR),2011 IEEE Conference on*. IEEE, 2011.

[2] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In *Computer Vision and Pattern*,*Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012.

[3] L. Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[4] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J.Lin. Liblinear: A library for large linear classification. *The* *Journal of Machine Learning Research*, 2008.

[5] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *12th European Conference* *on Computer Vision (ECCV)*. 2012.

[6] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision* *and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012.

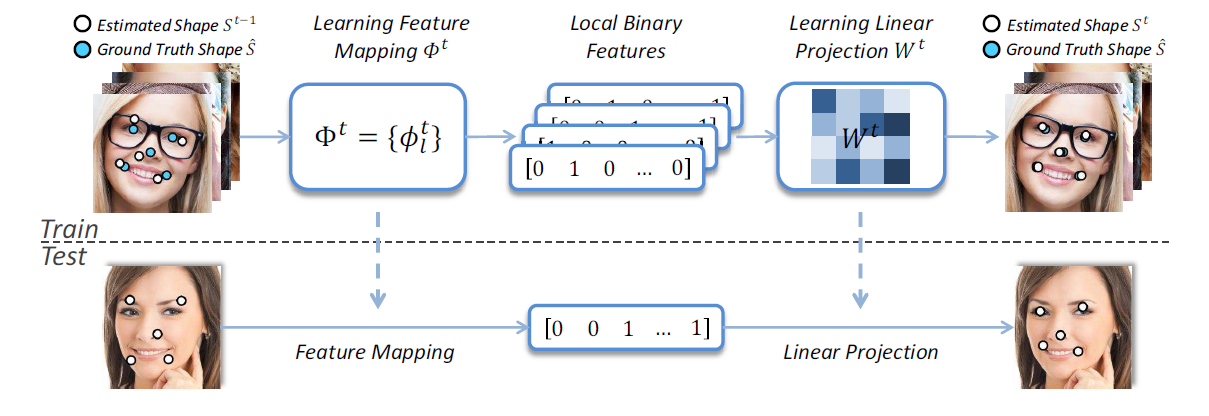
[7]. The extraction of image features of object detection <http://blog.csdn.net/zouxy09/article/details/7929531>

[8]. Reading Response <http://demo.netfoucs.com/boosting1/article/details/26085223>

[9]. Wikipedia- variance reduction <http://en.wikipedia.org/wiki/Variance_reduction>

**Technical description including algorithm:**

1. **Overview:**



All the story begin from this image and the formula (1).

For the above part of the image, as a training set, we hold their ground truth results, which are the position of all the landmarks , we can describe this as , and we initial the process with an initial shape . But only the information of positions of landmarks is not enough, because in each specific situation, we don’t know what to do, which direction we should transform it to, so we must learn – by local binary features.

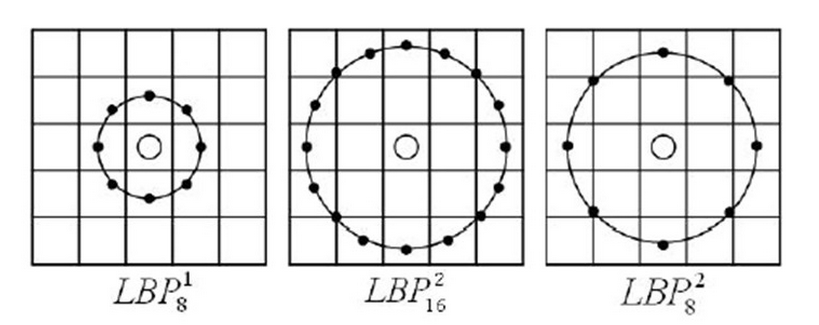
So we train a random forest to determine which kind pattern the landmark is. And do this procedure locally for each landmark using independent random forest. And then derive the concatenated local binary features, which is . After this part, we should determine how to transform, instead of transforming each landmark’s position by independent regression function, we choose the global regression.

The global regression, as global, means we’ll apply this matrix to the concatenated binary feature matrix and determine the offset of each landmark’s position, which is , the comes from the equation below:

After we get the delta shape , we add it to our former shape , such as the , so we get , it’s more closer to the ground truth shape, and we do all the thing again, after several iteration, the estimated shape will be infinitely close to the ground truth, in our experiment, it’s around 5 iterations.

1. **Local binary feature:**
   1. **Rotation and scale invariant**

We need a certain type of feature descriptor that will be invariant to scale , rotation , intensity and lighting.

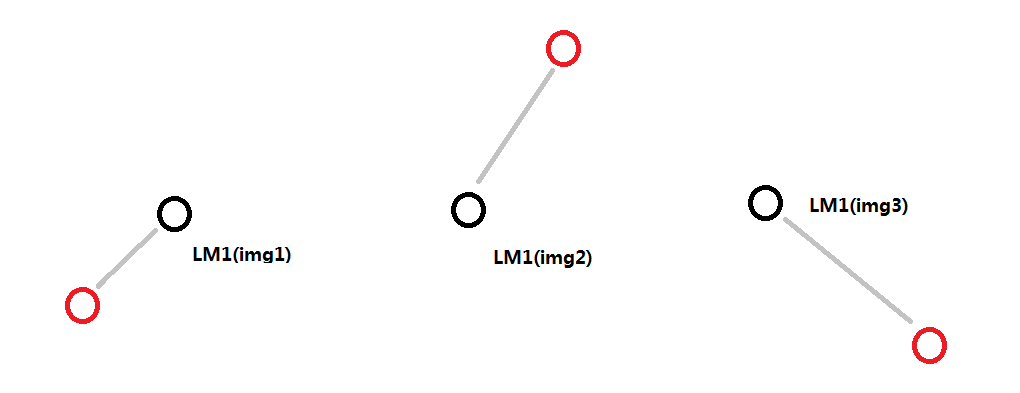


And the local binary pattern is the inspiration. It contains an angle and a radius, in the later procedure, we randomly produce pair of angle and radius to be scale and rotation invariant.

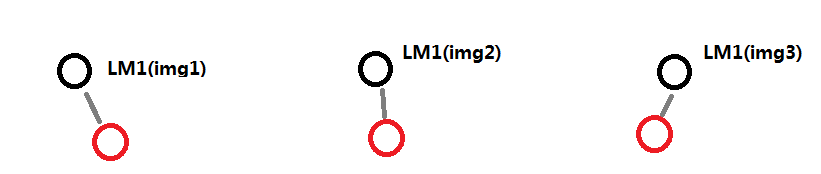
And by using the pixel-difference features to be intensity and lighting invariant. More detail in 3-random forest.

* 1. **locality principle and radius descending**

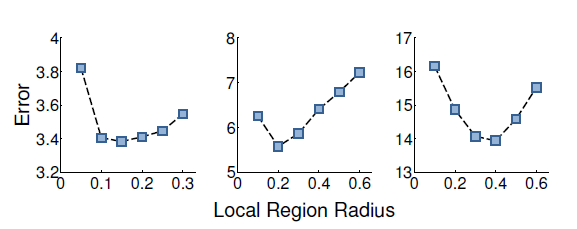
There are two important methods in feature learning, **1)we only consider the pixel features in the local region of a landmark. 2) we learn random forest for each landmark independently.  
 for 1)**, our goal is to predict the position offset of a landmark to update it to a more accurate position which is closer to the ground truth. But if the training dataset is drastically different, which means the  is widely scattered, then we need to use a large radiusto predict the offset, then each position can regress to the ground truth location quickly, like below.



But if the is clustered, a relatively smaller radius is more helpful to precisely predict the location and lower down the error.



Pic below is the statistics of this problem, and from the left to right, the are 0.05, 0.1, 0.2:



Even though, there is still some problem, because you should not keep the radius large, because after several regression, the position is pretty close to the ground truth, we should use smaller radius to ensure the precision, so in the algorithm, we gradually shrink the max radius to guarantee the prediction accuracy. Like the image below.



**For 2)**, we think although some research find the global learning will be better, but there are still some advantages for local learning each landmark independently.

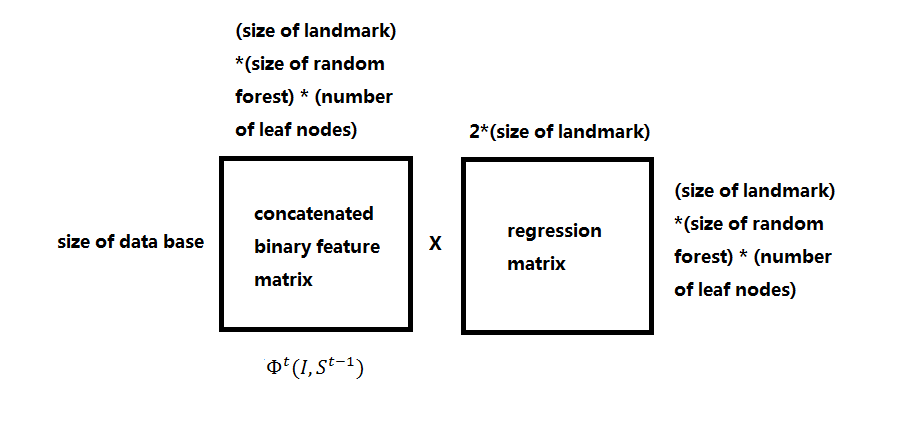
**First,** less noise, it’s obvious, because, on the contrary of global learning, which may find some features shared by multiple landmarks, it also import many useless noise for every landmark.

**Second,** although we use local learning, but as a compensation, we predict the regression globally.

**Third,** from the 1) part, we learn that, from the beginning, the radius is large, it may covers several neighbor landmarks and derive some shared features. And also lower down the computation budget.

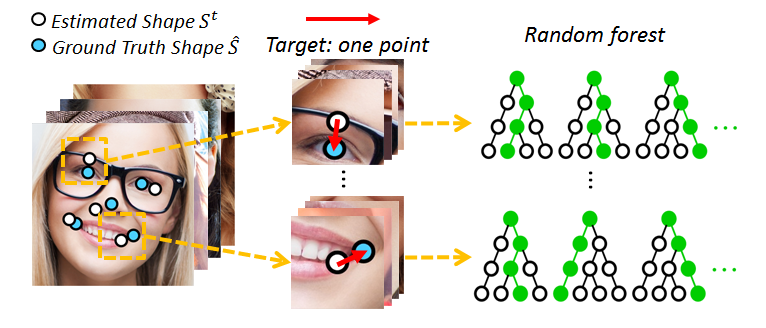
* 1. **concatenated binary feature**

After the computation of all the images in the database for every decision tree, we need to combine them into a concatenated binary feature matrix to train the regression function. Like below.



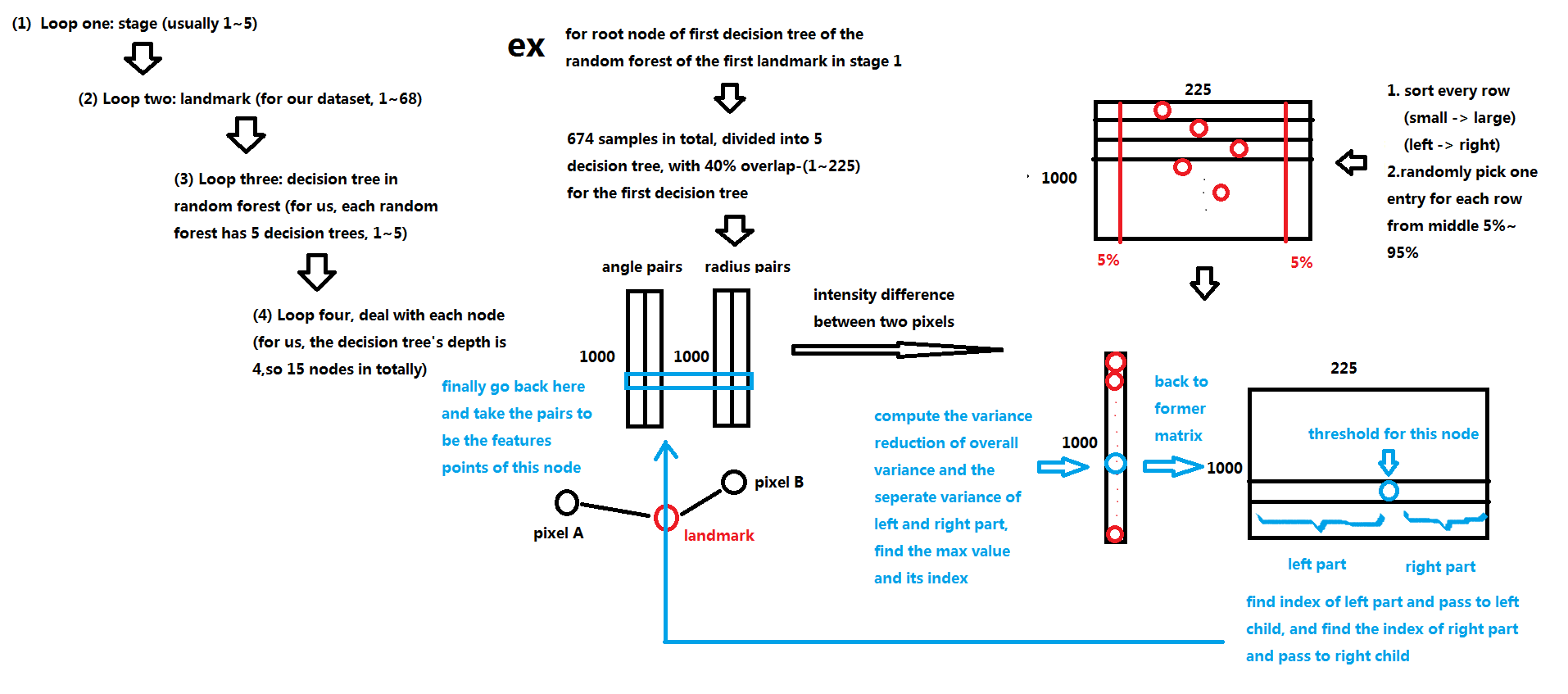
1. **Random forest**

After introduction to local binary feature, we need to form our random forest for each landmarks in each stage. The random forest will derive several decision using overlapped data sample but not exactly same to decide the local binary feature of each landmark for a image. Like the image below:



**3.1 how to derive the random forest?**

The whole loop looks like :



More details about:

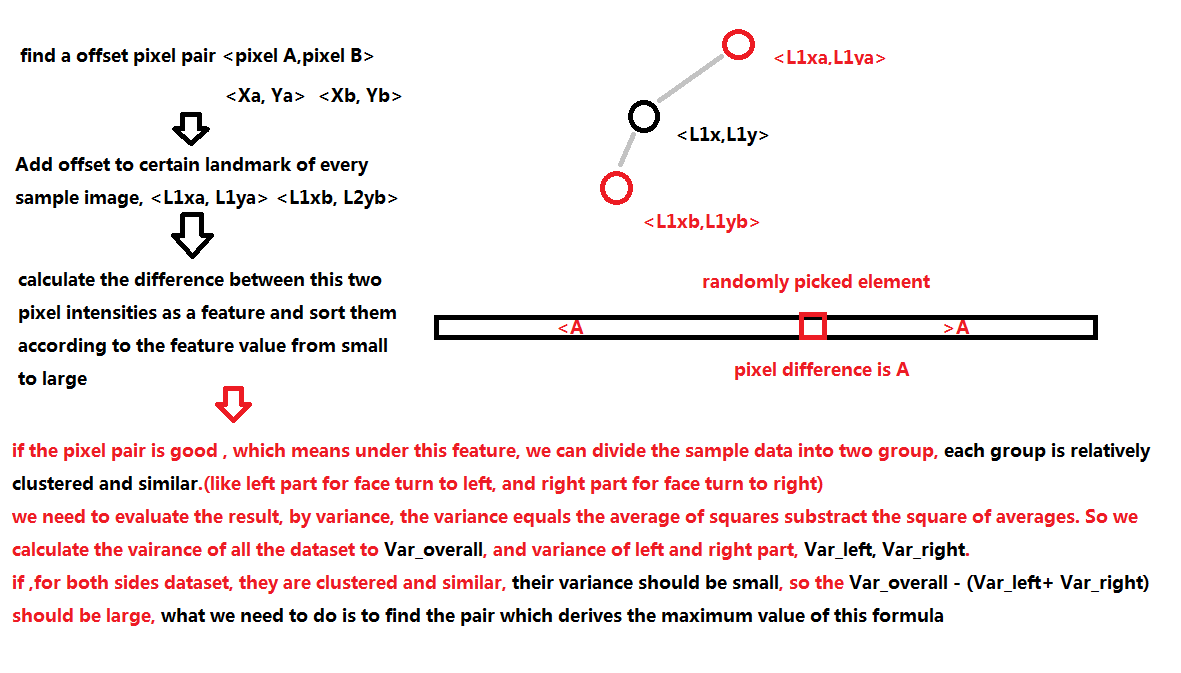
**1). The variance reduction:**

For the randomly picked element, we compute the variance reduction of the overall variance of residual and the sum of separate left and right part’s variance.

The residual is defined as:

: ground truth shape residual

We want the decision tree to divide the picture with clustered residual into one group, which means, for a specific node, if the data in left is similar and the same for right part, then the variances of left part and right part are both small , so the difference of overall variance between the sum of left and right variance is large.



**2). The pixel difference:**

As we said in section 2.1, we need a good feature descriptor, which is invariant to rotation, scale, lighting.

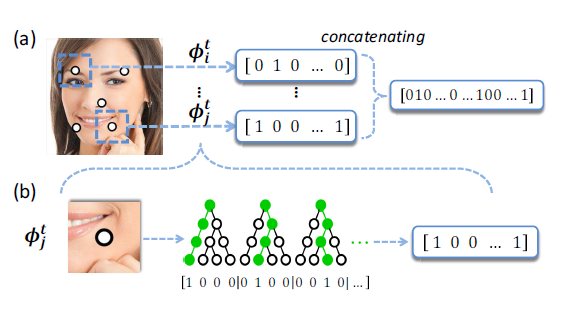
The randomly generated and tested 1000 pair solve the scale and rotation invariant problem, the pixel difference method deals with the lighting and exposure invariant problem.

**3). Depth , samples size, overlap and radius control**

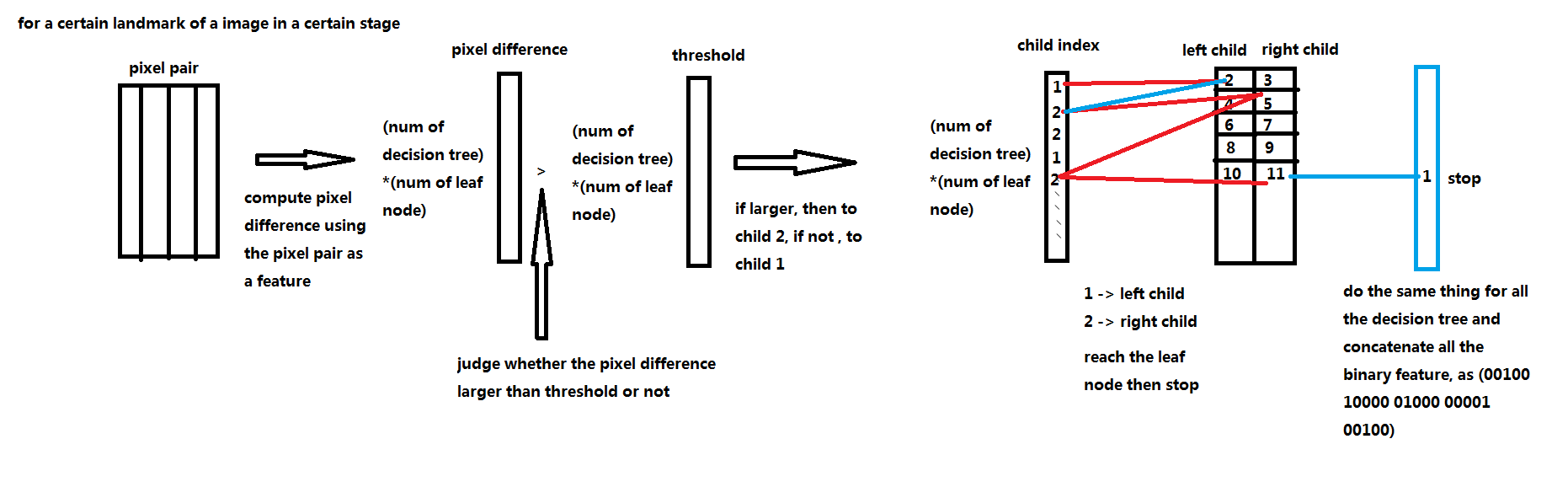
For different dataset, we should remain the control of the forest depth , sampling size ,overlap between decision trees and radius descending.

For our project, based on 68 landmarks in “afw” and “helen” image set, the depth 4 , sample size 1000~400, overlap around 0.4, radius from 0.4~0.05.

**3.2 derive binary features**

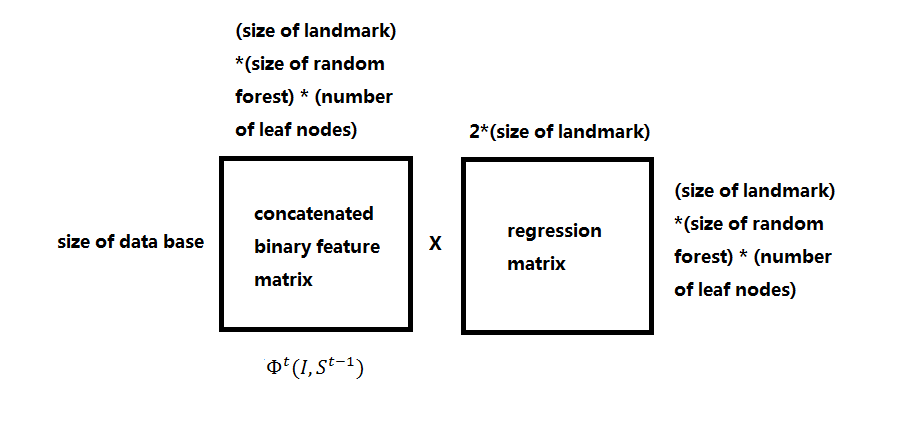
****

To derive and local binary feature of an image is actually following the random forest.



1. **Learning Regression function**

As we state before:

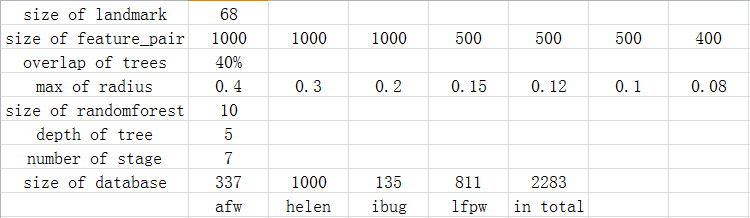


For this part, we are not going into detail, because we use the liblinear --- a library for large linear classification to get the .

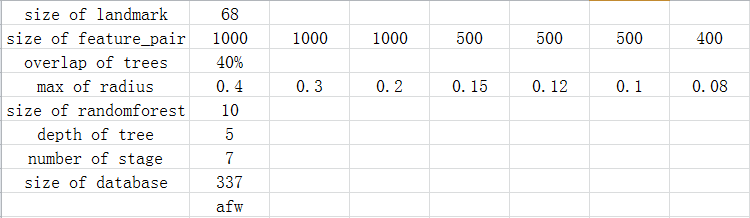
**Experiment result:**

We use **three** different experiment results to show the outcome of this paper:

1. **Training set for best performance----A**



1. **Rank two training set-----B**



1. **The third performance training set----C**



Then we use the distance of pupils to evaluate the error between test set alignment result and ground truth shape.

**Error in training :**

We didn’t give the error for training set A, because it’s very close to training set B, because the depth and size of random forest did not change so the error remains roughly same, only increase the discrimination ability for a more various different faces.

**Error for each test set:**

1).afw

2).helen

3).ibug

4).lfpw

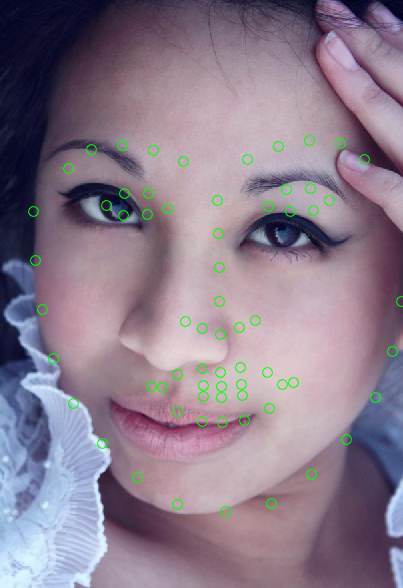
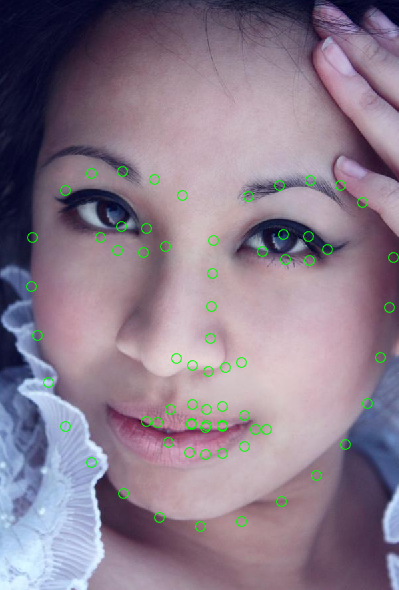
**Conclusion:**

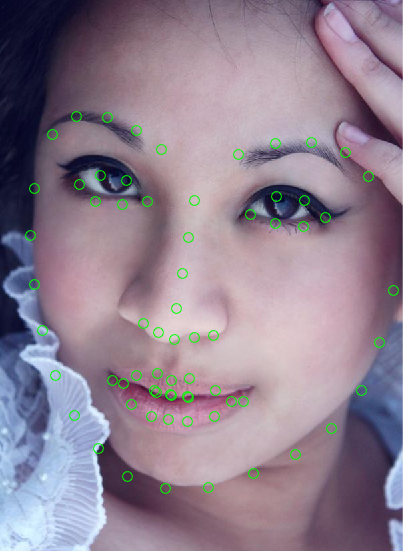
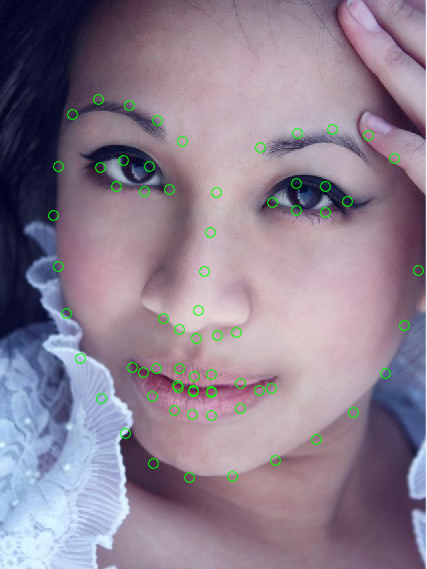
**There are some outlier data ,like theoretically speaking ,the A should perform better than B anyway, but in dataset faw, B actually win A, because we use afw to train B, so B perform better than A just in this case, we can see in other set, things back to reasonable.**

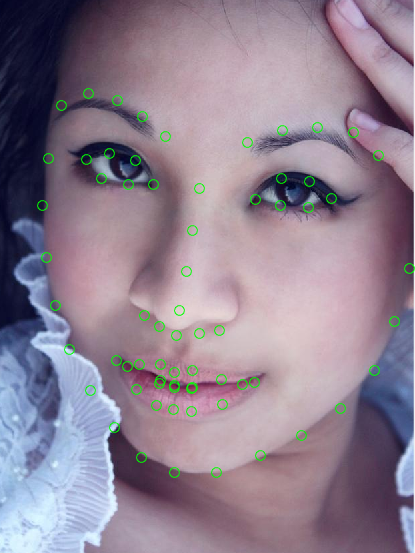
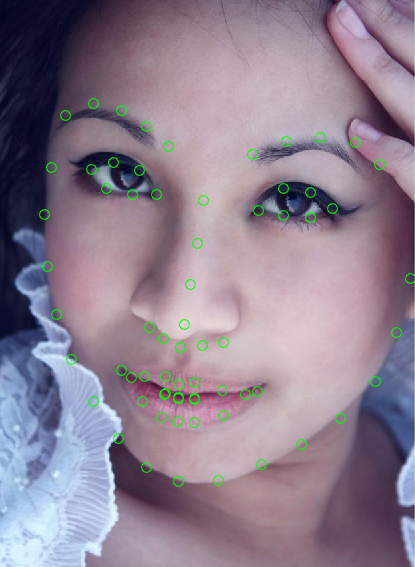
**And also we can observe that some curve tilt up in the end stage, because if the training set is not robust enough but we have stages more than a suitable number. The result will appears over-fitting. So just lower down the stage number will tackle that.**

**Result: (for the 2968560214\_1.jpg in helen)**

**1).for A:**

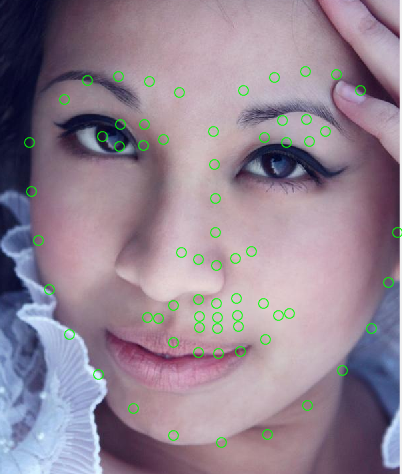
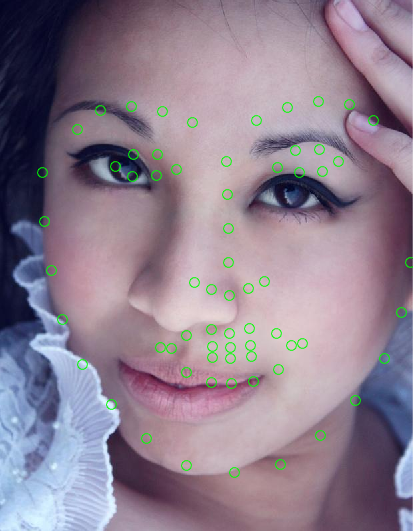
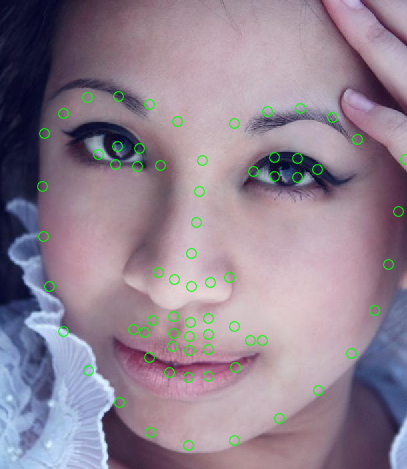
  

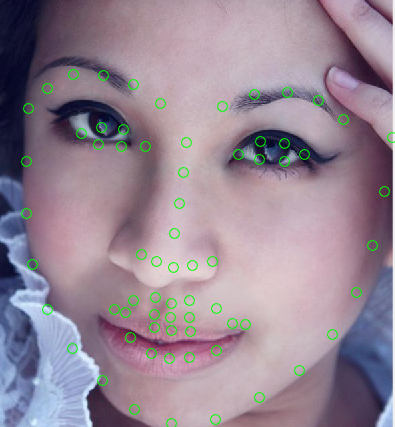
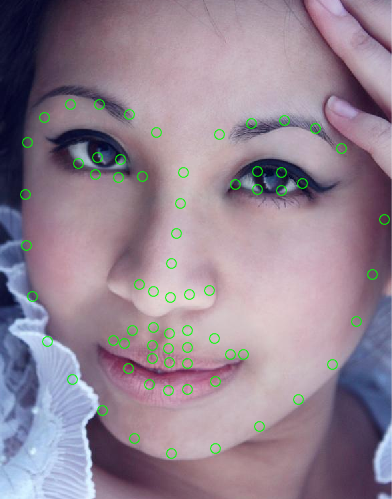
  

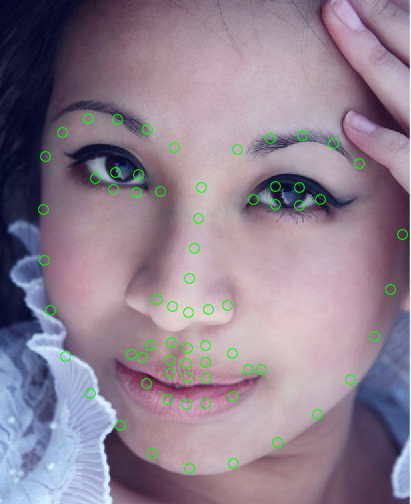
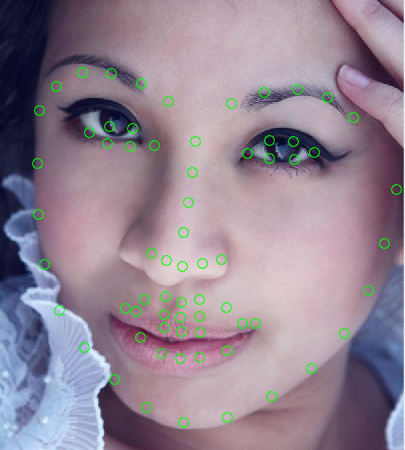
 

**For result A, it’s fabulou, perfectly close to ground truth.**

**2) for B:**

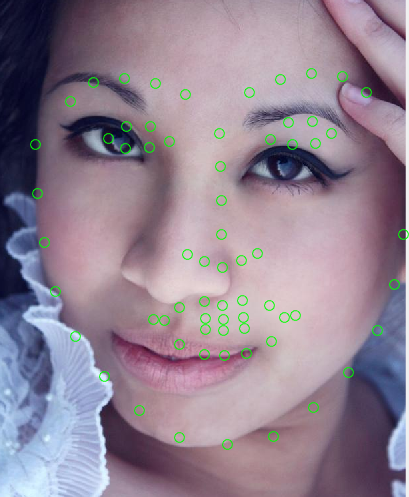
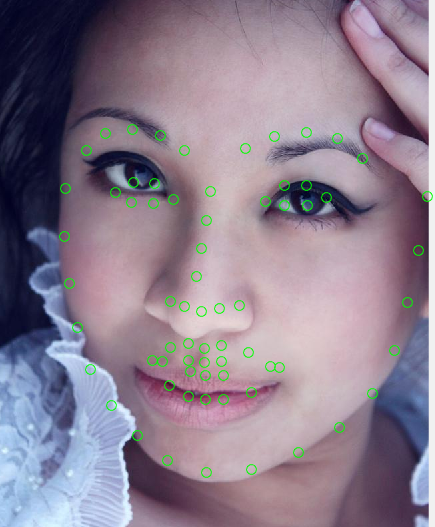
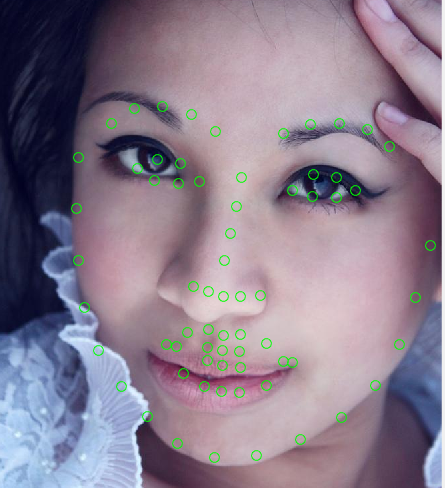
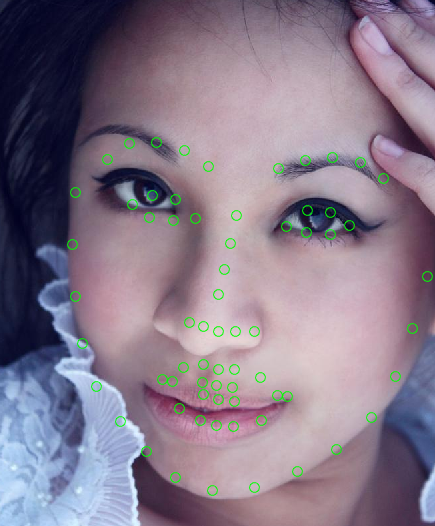
  

**We can see that because of the robustness of the training set is not good enough, so the final alignment did not perfectly match the ground truth.**

**And according to the error data, it’s overfitting in the last few stages.**

**3). For C**

**We can see that the depth and the size of random forest did not influence the outcome of the result, indicating that ,for such easy face, we do not need that much deep and wide forest but we need more training set to discriminate the feature of face pattern.**

**Discussion:**