Margareta Kušan 11.2.2016.

Face detectors

The point of this document is a brief review of the published methods of face detection listed on url from which the code is available online. The comparison between all these methods is given on the ROC curves: Dicrete, Continuous.

Face detectors with code available online:

1. NPDFace

S. Liao, A. Jain, S. Li. A Fast and Accurate Unconstrained Face Detector. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015. (legend: NPDFace)

website: http://www.cbsr.ia.ac.cn/users/scliao/projects/npdface/index.html - MATLAB and C++, Windows and Linux

paper: http://www.cbsr.ia.ac.cn/users/scliao/papers/Liao-PAMI15-NPD.pdf

Their goal was to develop effective features and robust classifiers for unconstrained face detection with arbitrary facial variation. They introduce the use of simple pixel-level feature, called the Normalized Pixel Difference (NPD). An NPD is computed as the ratio of the difference between any two pixel intensity values to the sum of their values. They also propose a deep quadratic tree learning method and construct a single soft-cascade AdaBoost classifier to handle complex face manifolds and arbitrary pose and occlusion conditions. They claim their algorithm is fast, resistant to occlusions, pose variations, illumination and blur or low image resolution.

In the paper they give the table of testing the speed of the detection on a two minute length video with 3 different resolutions as opposed to openCV face detector (number 10. on this list).

CPU	Resolution	NPD	OpenCV
Atom N450 @1.6GHz (1 core, 2 threads)	640 × 480	**19.4**	2.1
	800 × 600	**12.1**	1.3
	1280 × 720	**6.8**	0.7
	1920 × 1080	**3.0**	0.3
i5-2400 @3.1GHz (4 cores, 4 threads)	640 × 480	**177.6**	24.4
	800 × 600	**112.6**	16.2
	1280 × 720	**63.3**	8.9
	1920 × 1080	**29.6**	3.6

On the website they allow download of detector code and the trained detectors. Also they provide training code.

2. MultiresHPM

G. Ghiasi, C. Fowlkes. Occlusion Coherence: Detecting and Localizing Occluded Faces. Technical Report, June 2015 [arXiv:1506.08347]. (legend: MultiresHPM)

github: https://github.com/golnazghiasi/hpm-detection-code, MATLAB with precompiled Mex files for Linux included.

paper: http://arxiv.org/pdf/1506.08347.pdf

Hierarchical deformable part model for face detection and keypoint localization that explicitly models part occlusion. It is stated that his two-layer model provides a compact, discriminative representation for the appearance and deformations of parts. It also captures the correlation in shapes and occlusion patterns of neighboring parts (e.g., if the chin is occluded it would seem more likely the bottom half of the mouth is also occluded). In addition to representing the face shape, each part has an associated occlusion state chosen from a small set of possible occlusion patterns, enforcing coherence across neighboring keypoints and providing a sparse representation of the occluder shape where it intersects the part.

In images with relatively little occlusion, the HPM gives similar detection and localization performance to other part-based approaches, e.g. DPM, but is significantly more robust to occlusion.

It is based on the work of Zhu and Ramanan (7. on this list) and also this paper: url .

3. DDFD

S. S. Farfade, Md. Saberian and Li-Jia Li. Multi-view Face Detection Using Deep Convolutional Neural Networks. International Conference on Multimedia Retrieval (ICMR), 2015. (legend: DDFD)

github: https://github.com/guoyilin/FaceDetection_CNN - Python with Caffe library.

paper: http://arxiv.org/pdf/1502.02766v3.pdf

Neural network based classifier.

Relevant from the paper: 'Deep Dense Face Detector (DDFD) is a method that does not require pose/landmark annotation and is able to detect faces in a wide range of orientations using a single model based on deep convolutional neural networks. The proposed method has minimal complexity; unlike other recent deep learning object detection methods, it does not require additional components such as segmentation, bounding-box regression, or SVM classifiers. Furthermore, we analyzed scores of the proposed face detector for faces in different orientations and found that 1) the proposed method is able to detect faces from different angles and can handle occlusion to some extent, 2) there seems to be a correlation between distribution of positive examples in the training set and scores of the proposed face detector.'

4. Pico

N. Markus, M. Frljak, I. S. Pandzic, J. Ahlberg and R. Forchheimer. A Method for Object Detection Based on Pixel Intensity Comparisons Organized in Decision Trees. CoRR 2014. (legend: Pico)

github: https://github.com/nenadmarkus/pico - Provides code to learn your own detector, written in Python 3, has tutorial on the process of learning your own face detector.

paper: http://arxiv.org/pdf/1305.4537v5.pdf

Cascade based detector, a modification of the standard Viola-Jones object detection method.

The basic idea is to scan the image with a cascade of binary classifiers at all reasonable positions and scales. An image region is classified as an object of interest if it successfully passes all the members of the cascade. Each binary classifier consists of an ensemble of decision trees with pixel intensity

comparisons as binary tests in their internal nodes. This enables the detector to process image regions at very high speed.

Some highlights of pico are:

- High processing speed.
- There is no need for image preprocessing prior to detection.
- There is no need for the computation of integral images, image pyramid, HOG pyramid or any other similar data structure.
- All binary tests in internal nodes of the trees are based on the same feature type (not the case in the V-J framework).
- The method can easily be modified for fast detection of in-plane rotated objects.

5. HeadHunter

M. Mathias, R. Benenson, M. Pedersoli and L. Van Gool. Face detection without bells and whistles. ECCV 2014. (legend: HeadHunter)

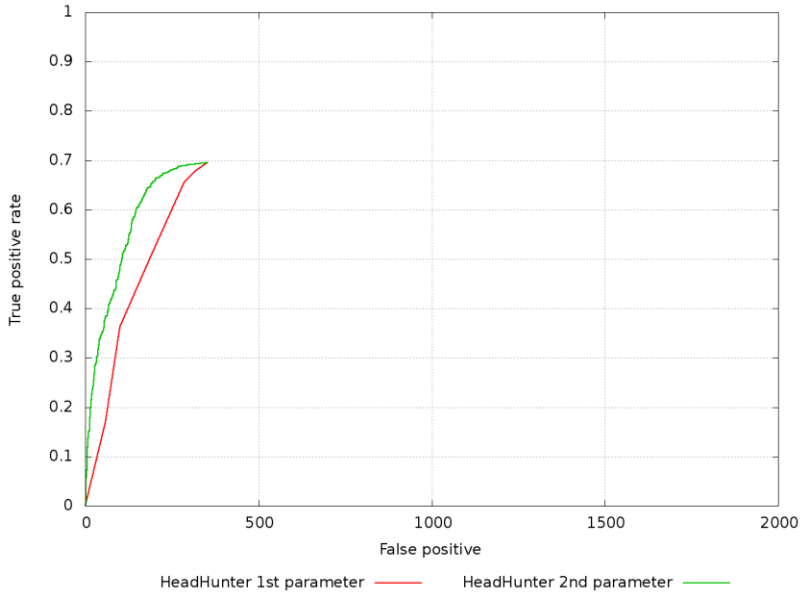
paper:
http://rodrigob.github.io/documents/2014_eccv_face_detection_with_supplementary_material.pdf

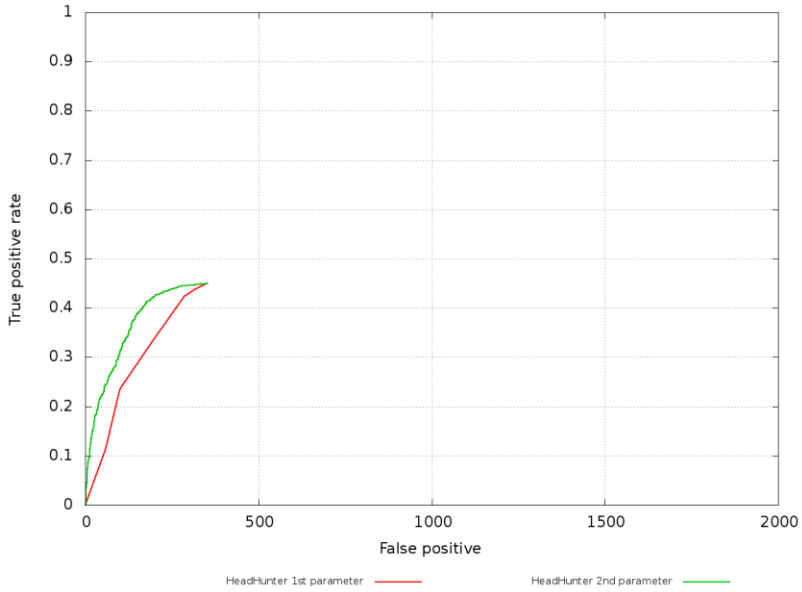
code: can be found as a part of this project https://bitbucket.org/rodrigob/doppia

Integral channel features model using small set of rigid models (without deformable parts).

In their work, they conclude that properly trained DPM baseline obtains top performance across all datasets considered – but there is a big importance to a baseline design , the importance to low level details (a single threshold value makes the difference between under-performing to top-performing) and the value of open source release of research material. Besides DPM, the HeadHunter (a set of rigid templates) essentially reaches the same performance. That indicates that parts are useful but not critical to reach top performance. As long as enough training data is available to cover pose diversity, a small set of rigid templates will detect faces as good as anything else.

ROC curves that I obtained using their normal output with bounding boxes (first one is continuous, second one is discrete) :





6. Joint cascade

D. Chen, S. Ren, Y. Wei, X. Cao, J. Sun. Joint Cascade Face Detection and Alignment. ECCV 2014. (legend: Joint Cascade)

github: https://github.com/luoyetx/JDA

paper: http://home.ustc.edu.cn/~chendong/JointCascade/ECCV14_JointCascade.pdf

From the paper: ‘ The key idea is to combine face alignment with detection, observing that aligned face shapes provide better features for face classification. To make this combination more effective, our approach learns the two tasks jointly in the same cascade framework, by exploiting recent advances in face alignment. Such joint learning greatly enhances the capability of cascade detection and still retains its realtime performance. Extensive experiments show that our approach achieves the best accuracy on challenging datasets, where all existing solutions are either inaccurate or too slow.’

From github: ‘ It's a shame that I still not get a fully trained model now. There is many detailed problems of JDA algorithm which need to be discussed. ‘

This detector takes only 28.6 milliseconds for a VGA image, more than 1000 times faster than the next detector on this list (Zhu et all), and about the approximated time as Viola-Jones! In the runtime, detector needs only 15MB memory which is significantly less than other methods.

7. X. Zhu and D. Ramanan. Face detection, pose estimation and landmark localization in the wild. CVPR 2012.

website: https://www.ics.uci.edu/~xzhu/face/ , MATLAB. They also offer core to train your own models.

paper: http://www.ics.uci.edu/~xzhu/paper/face-cvpr12.pdf

It is based on a mixtures of trees with a shared pool of parts; they model every facial landmark as a part and use global mixtures to capture topological changes due to viewpoint. They claim that tree-structured models are surprisingly effective at capturing global elastic deformation, while being easy to optimize unlike dense graph structures.

Disadvantages:

- Slow compared to Viola Jones - ~10 seconds on VGA quality image (640x480).
- Tuned for large faces - larger than 80x80 pixels.

It is trained on large faces, so if landmark is not a main concern, and the detector will be used in *detecting small faces*, they recommend training your own model on smaller faces.

8. SURF

J. Li, T. Wang and Y. Zhang. Face Detection using SURF Cascade. ICCV 2011 BeFIT workshop.

github: https://github.com/xieguotian/SurfFaceDetection C++, Visual Studio project.

A lot of information available in these slides:
http://face.cs.kit.edu/befit/workshop2011/pdf/slides/jianguo_li-slides.pdf.

Cascade based face detector. It's stated that the source code is not finished yet, the model training source code is not published yet.

9. Viola-Jones OpenCV

This is a standard face detection first object detection framework to provide competitive object detection rates in real-time proposed in 2001 by Paul Viola and Michael Jones. Lots of information about this face detector can be found online. The standard implementation of Viola-Jones face detector algorithm is in the OpenCV library.

Disadvantages of Viola–Jones algorithm:

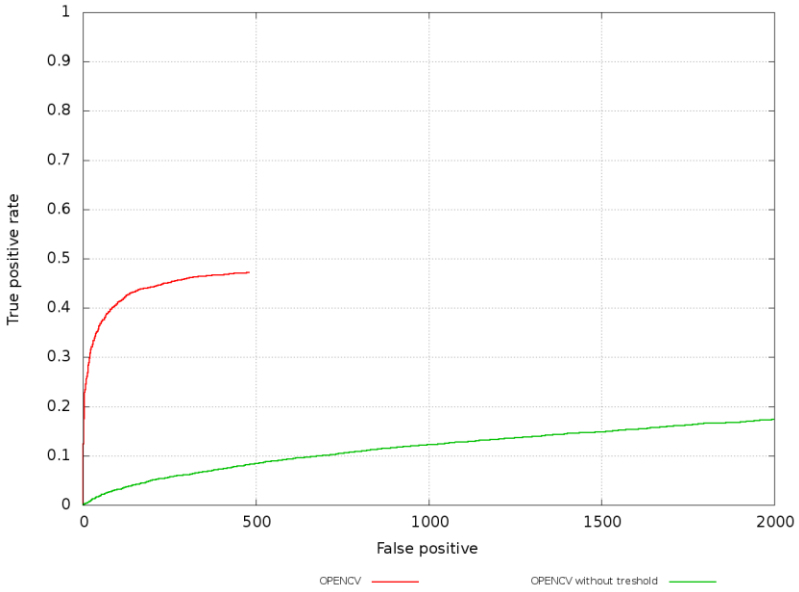
- Detector is most effective only on frontal images of faces
- It can hardly cope with 45° face rotation both around the vertical and horizontal axis.
- Sensitive to lighting conditions
- We might get multiple detections of the same face, due to overlapping sub-windows.

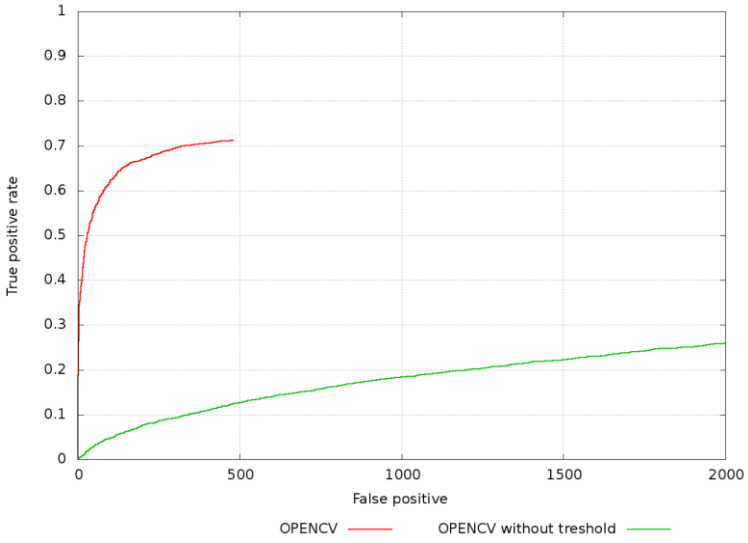
Detection using OpenCV 3.0, with function detectMultiscale, using 'haarcascade_frontalface_default.xml' and parameters :

- scaleFactor=1.2,

- minNeighbors=5,
- minSize=(30, 30)

and detection with minNeigbours=0 (removes the threshold which gives us all initial squares):





10. W. Kienzle, G. Bakir, M. Franz and B. Scholkopf Face Detection - Efficient and Rank Deficient.

Advances in Neural Information Processing Systems, 2005.

website: http://people.kyb.tuebingen.mpg.de/kienzle/fdlib/fdlib.htm, for C/C++ or MATLAB.

There is no paper following the code of this detector. Works really fast but has no output detection scores. One idea for providing an output score is running it at different tresholds and checking which bounding boxes will stay.

11. K. Mikolajczyk, C. Schmidt and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. ECCV 2004.

website: http://www.robots.ox.ac.uk/~vgg/research/affine/face_detectors.html - The detector works on pgm and ppm images. Works really fast.

Detector based on a probabilistic assembly of robust part detectors. It does not only detect faces, but it is built to detect a whole body. At the time (2004.) it was the best suggested face detector, but today a lot of other methods have been developed that outperform this detector.

Conclusion

Face detection is still a subject of great interest and with room for improvement, it is a fast developing field. With that being said, if I were to decide which detector from the list I would include into my work, I would definitely concentrate on the newest detectors (DDFD, NPDFace, MultiresHPM, Joint Cascade). Out of these detectors, depending on the problem I am trying to solve I would choose the most important features of my interest. With the face detection in videos, speed in FPS would be one of the most important features. That is where NPDFace and Joint Cascade claim to have the best results. However, looking at the evaluation using FDDB benchmark, MultiresHPM and Joint Cascade seem to give the best results out of these detectors. So it would be logical to decide between these detectors based on the desired ratio of the detector quality and speed. Also, there is a feature of occlusion that has to come into the equation, for example MultiresHPM claims to be very robust to occlusions.