# Feature Extraction

The authors have presented a novel technique for detecting face liveness in the paper. This approach tries to extract features pertaining to image quality on multiple scales. Their proposal provided a manual feature engineering approach measuring 21 features. The following section delves deep into how each of them is calculated.

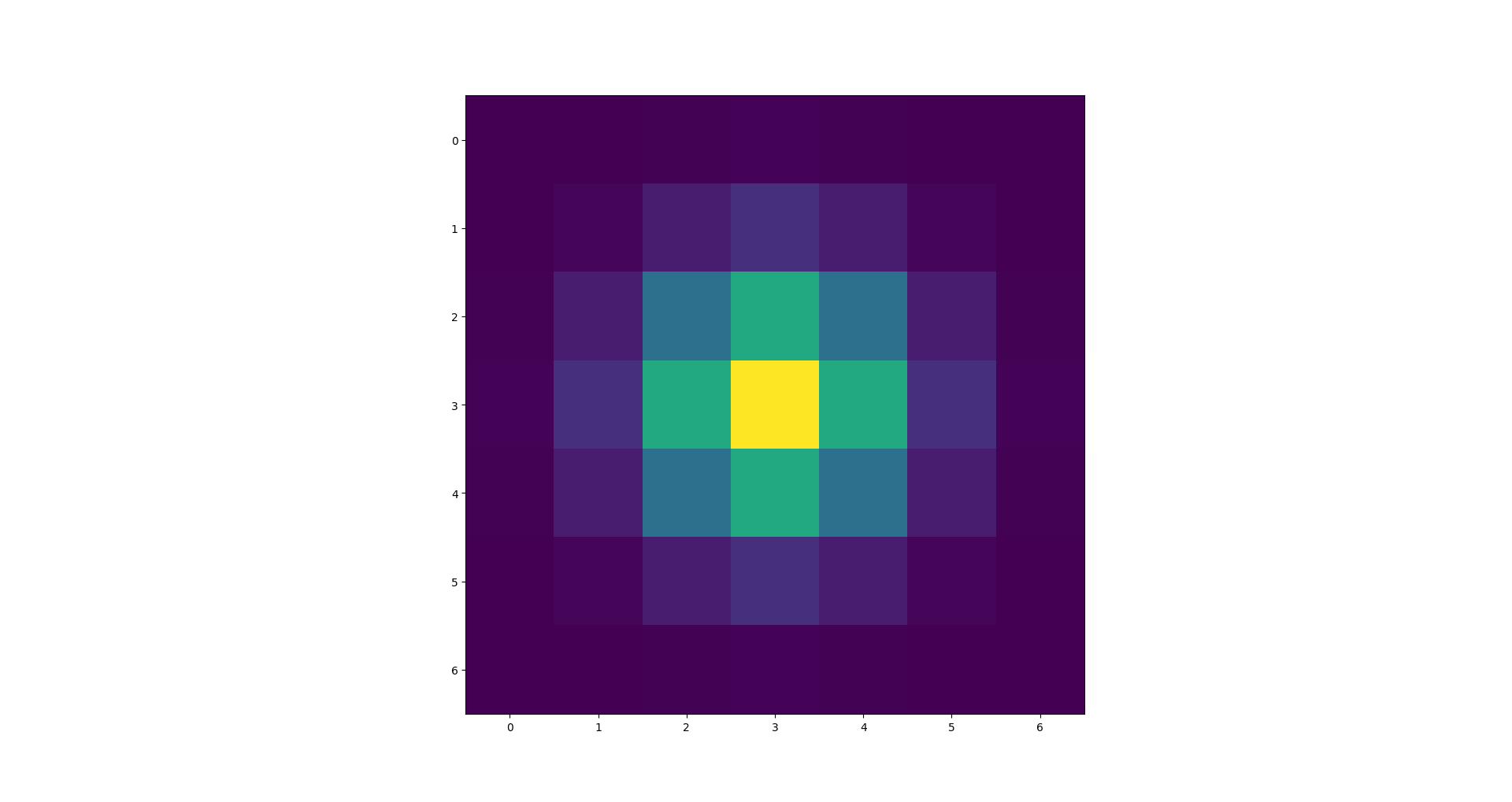
## Features 1-2

Mean subtracted and contract normalized (MSCN) coefficients [RH2] are very sensitive to image quality. Authors are using modified MSCN values to employ it for face liveness detection. The same is calculated as follows:

Here, is the grayscale face image obtained from section [cite]. Assuming it’s shape is M x N. Also, and are local image coordinates, and . and are local mean and local standard deviation matrices with the same shape as . Those are defined as follows:

Here, a 2D Gaussian kernel that is zero-centered and has a standard deviation of 1.17 in horizontal and vertical directions. has a shape of 7 x 7 as it’s sampled up to three sigma to cover 99.7% of probability mass. Such is also normalized to have unit volume.

It can be seen below in Fig H1,



H1

is MSCN coefficient matrix. Since the Gaussian window is sized 7 x 7, the face image is edge padded first before calculating so that it retains the same shape as , that is M x N. The values of MSCN matrix follow a bell-shaped distribution as you can see below in Fig H4,



Fig H2: A frame from the fake video



Fig H3: Detected face from the image shown in fig H2 and converted to grayscale

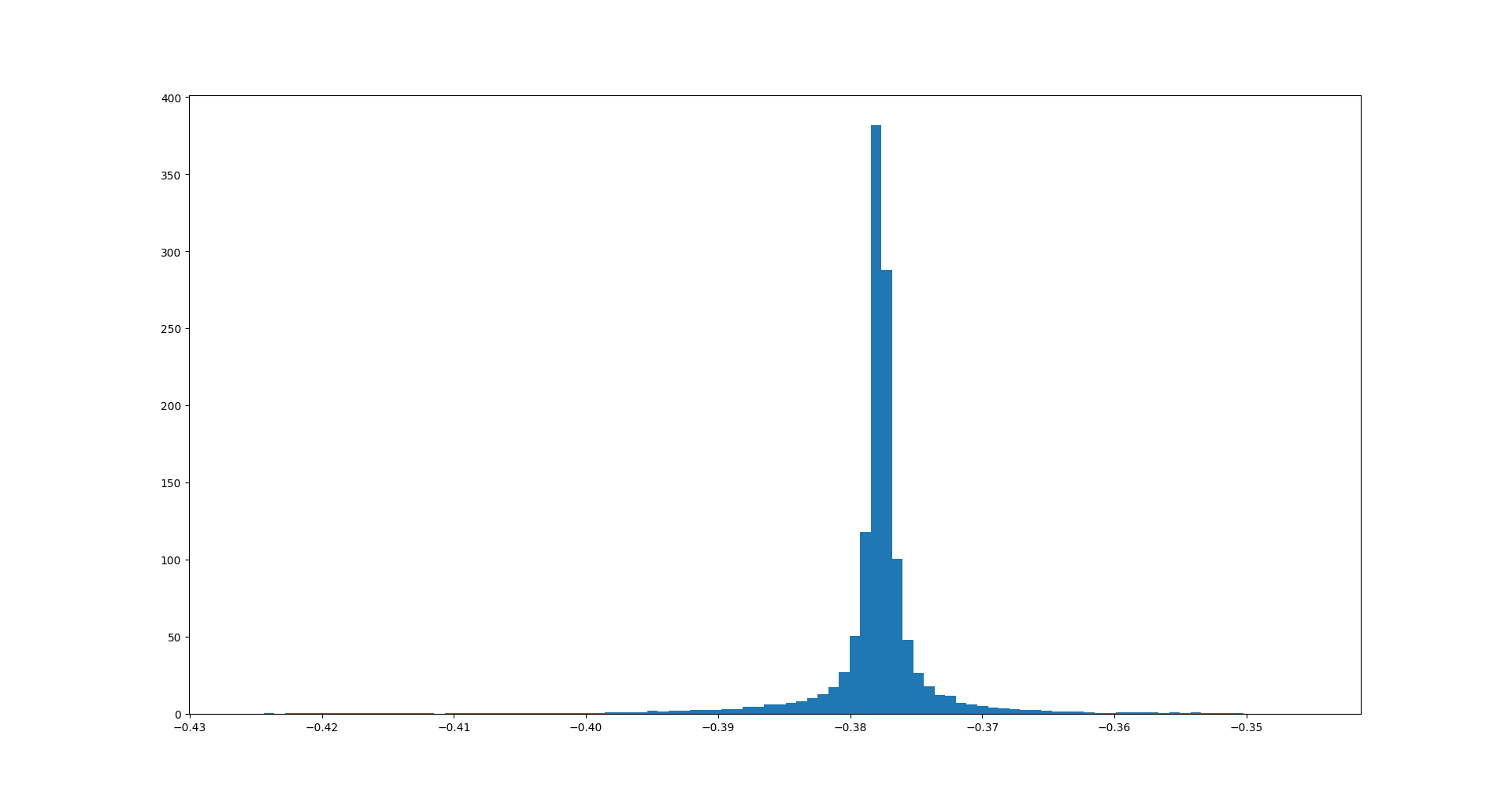


Fig H4: Normalized histogram of MSCN coefficients calculated from the image shown in Fig H3.

The researchers suggest estimating this distribution using Symmetric Generalized Gaussian Distribution (SGGD), a probability density function (PDF). It’s defined as follows:

Where and

Variance of this distribution is given by,

The parameters and are estimated by minimizing the sum of negative log-likelihood of this function on observed data points, which in this case is our MSCN coefficients. Such is done using the SciPy Python library. and form two features out of 21. The estimated SGGD distribution for MSCN coefficient distribution is shown in Fig H4 below.

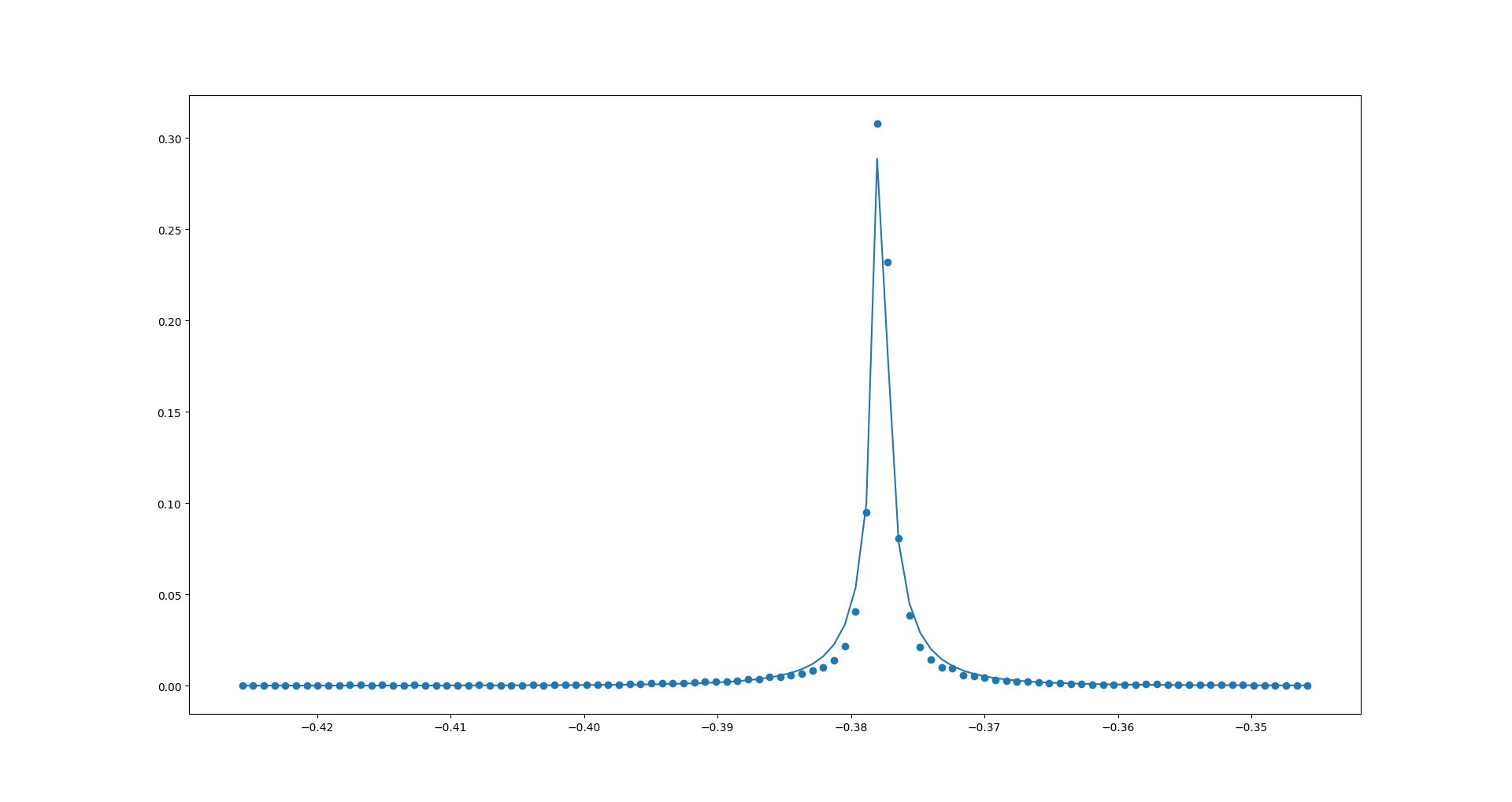


Fig H5: Estimated SGGD (line plot) imposed on the normalized histogram of MSCN coefficients (dotted/scatter plot)

The dotted/scatter plot indicates the normalized histogram of the MSCN coefficients, and the line plot refers to the estimated SGGD distribution.

## Features 3-10

The paper describes calculating these eight features by following [RH3]. It instructs to find the product of two adjacent MSCN coefficients. There are a total of eight such pairs, which are shown below.

| |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | | |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | | |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | | |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | | |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | | |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | | |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | |

The center MSCN coefficient is multiplied with eight neighboring coefficients to form eight adjacent MSCN matrices, each represented by where . Before calculating , is padded such that has shame shape as which is M x N.

In the paper, the normalized histogram of each is estimated using Asymmetric Generalized Gaussian Distribution (AGGD), a probability density function (PDF). It’s used in modeling skewed data distributions. It’s defined as follows:

Where .

Here the parameters determine the skewness in the data. Such can be represented by

To estimate these parameters, the sum of negative log-likelihood of this function must be minimized. Minimization is usually done by calculating the partial derivatives with respect to the PDF’s parameters. Differentiating a piecewise function such as the above is tricky, even for libraries like SciPy. Doing the same in SciPy gave inaccurate results. Using these results for further calculations might compromise the integrity of this work. One of the examples is shown below in Fig H6.

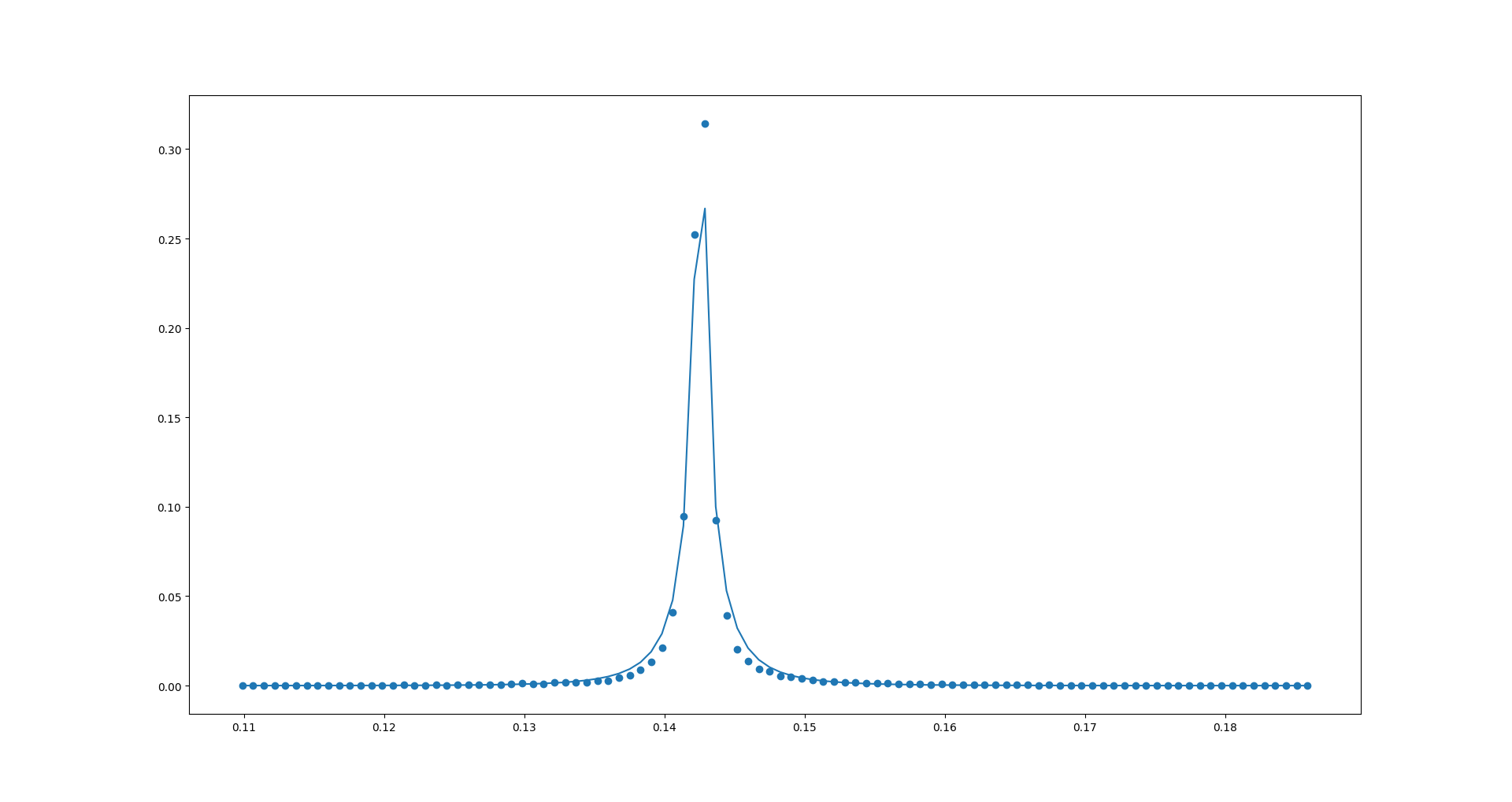


Fig H6: (Left) Input face grayscale image, (Right) Dotted plot refers to the normalized histogram of of the left image, and line plot refers to estimated AGGD distribution. The matching around the center is inaccurate. That area accounts for most of the probability mass.

We’re estimating the Skewed Voigt Distribution (SVD) to circumvent this issue. Doing so gave us better results, as seen in Fig H7. The skewed Voigt function is described below.

Where

and

is the complementary error function and is the error function.

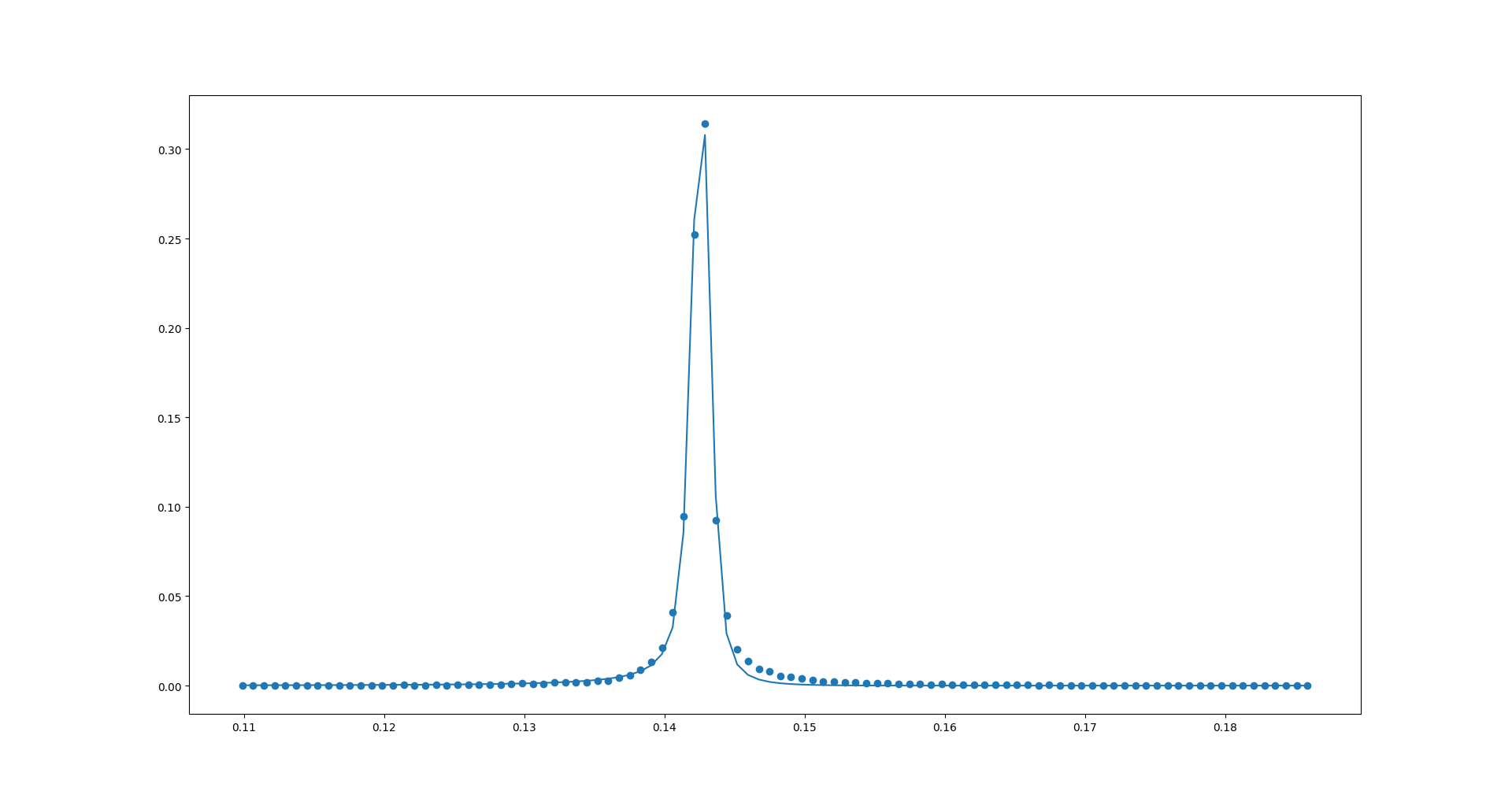


Fig H7: Estimated Skewed Voigt Distribution for the normalized histogram of of the left image shown in Fig H6.

For each , the skew parameter is estimated. Since there are eight such matrices, there are eight skew values which form our feature 3 to feature 10.

## Features 11-20

Furthermore, the authors propose to calculate a downsampled image . The grayscale face image is downsampled by two and blurred using the 2D Gaussian kernel shown in Fig H1 to obtain . Repeating the steps shown in Section [] and [] on image , form feature 11 to feature 20.

## Feature 21

Here, the researchers propose to measure a gradient-based feature to strengthen the feature set further for the face liveness detection task. Firstly, the proposed technique instructs calculating distorted image by applying a 2D Gaussian Kernel with standard deviation = 0.6 on .

Afterwards, the gradient magnitude of I and D are calculated using horizontal and vertical Prewitt kernels shown below. The gradient magnitude of I and D are called and respectively. It has same shape as the input image, which is M x N.

Afterwise the Gradient Magnitude Similarity (GMS) denoted as S is calculated as follows:

Here, is a small number used for stabilization purposes.

S is further altered using the following criteria.

Change to 0 if it’s less than the threshold . Obtained is now called Effective Pixel Similarity (EPS) map. The EPS map corresponding to Fig H6 left is shown below in Fig H8. Here, white pixels only passed the aforementioned criteria.

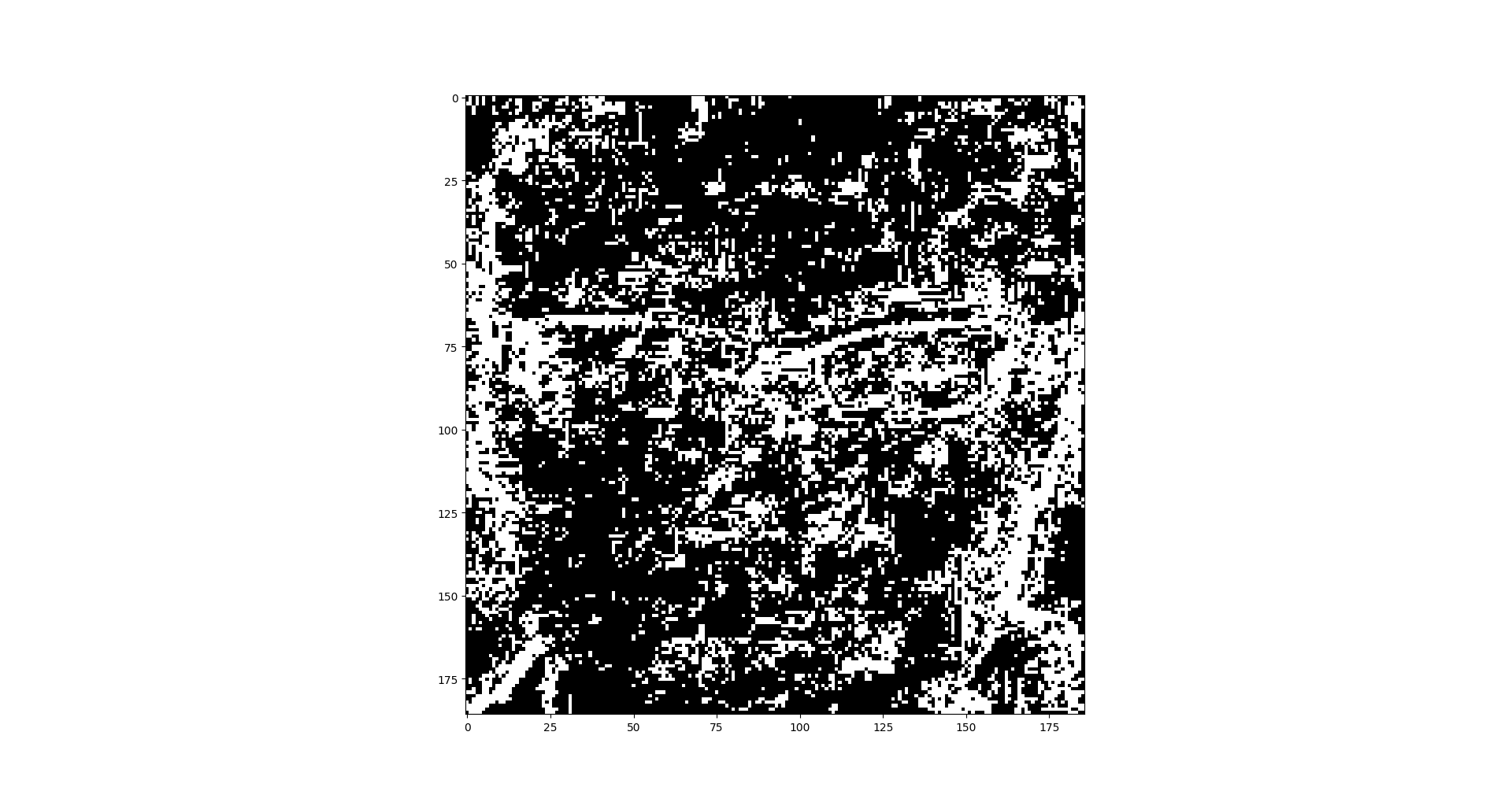


Fig H8

Furthermore, the average value of the EPS map is calculated, denoting it as . Finally, the last feature is measured using the following equation. Such concludes the feature extraction section.

## References

RH2. D. L. Ruderman and W. Bialek, "Statistics of natural images: Scaling in the woods," in Advances in neural information processing systems, 1994, pp. 551-558.

RH3. A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," IEEE Transactions on Image Processing, vol. 21, no. 12, pp. 4695-4708, 2012.