

Fingerprint Quality Validation

Lưu Nam Đạt

Mar-28-2023

Part 1: Variance of Gray Levels

$$mean(I) = \mu = \frac{1}{N} \times \sum_{i=1}^N E_i$$

$$var(I) = \sigma^2 = \frac{\sum_{i=1}^N (E_i - \mu)^2}{N}$$

$$std(I) = \sigma = \sqrt{var(I)}$$

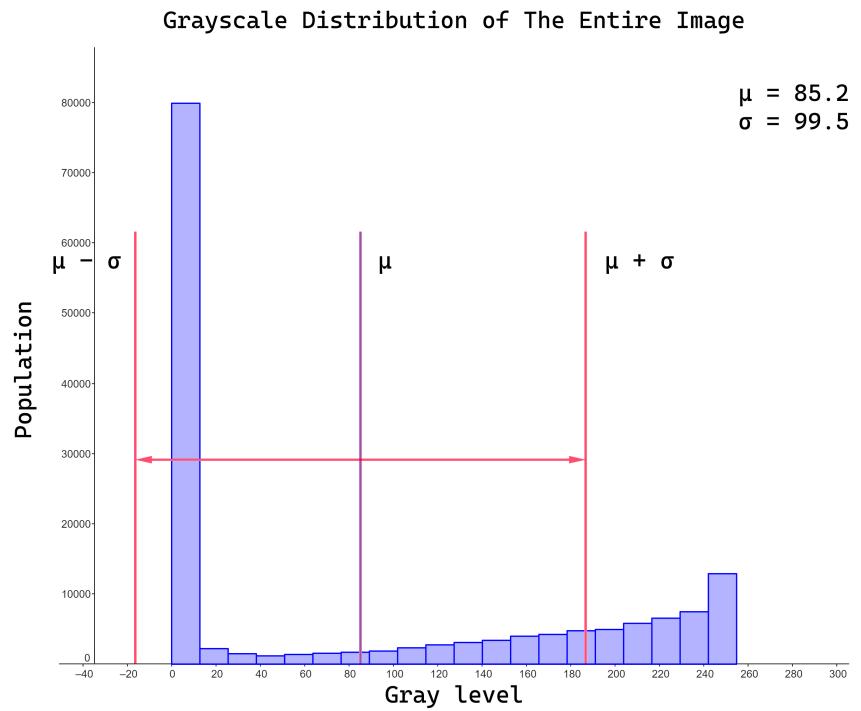


Figure 1: Grayscale Distribution of a Fingerprint Image

Let σ_{base} be the standard deviation of the image

The contrast quality (cq) of a block β is determined by:

$$cq_\beta = \frac{\sigma_\beta}{\sigma_{base}}$$

High cq_β value means that the block β contains both clear ridges and clear valleys, which promises useful data.

If cq_β is too low, β can either be a background block, or a block without any helpful information at all (bad block).

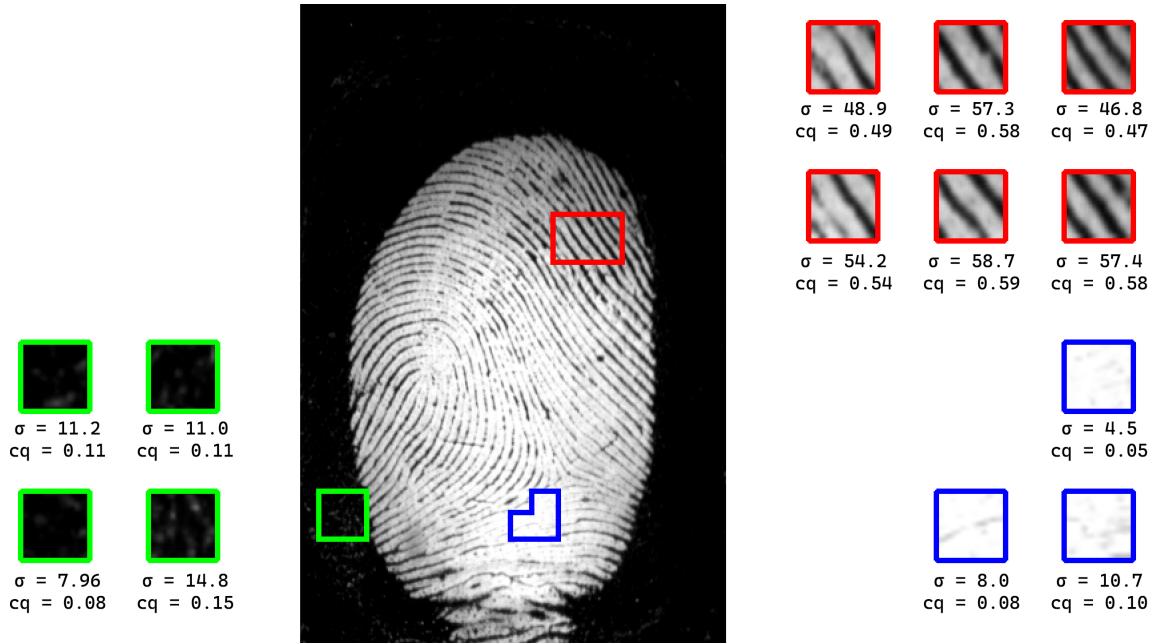


Figure 2: Standard Deviation and Contrast Quality of some blocks

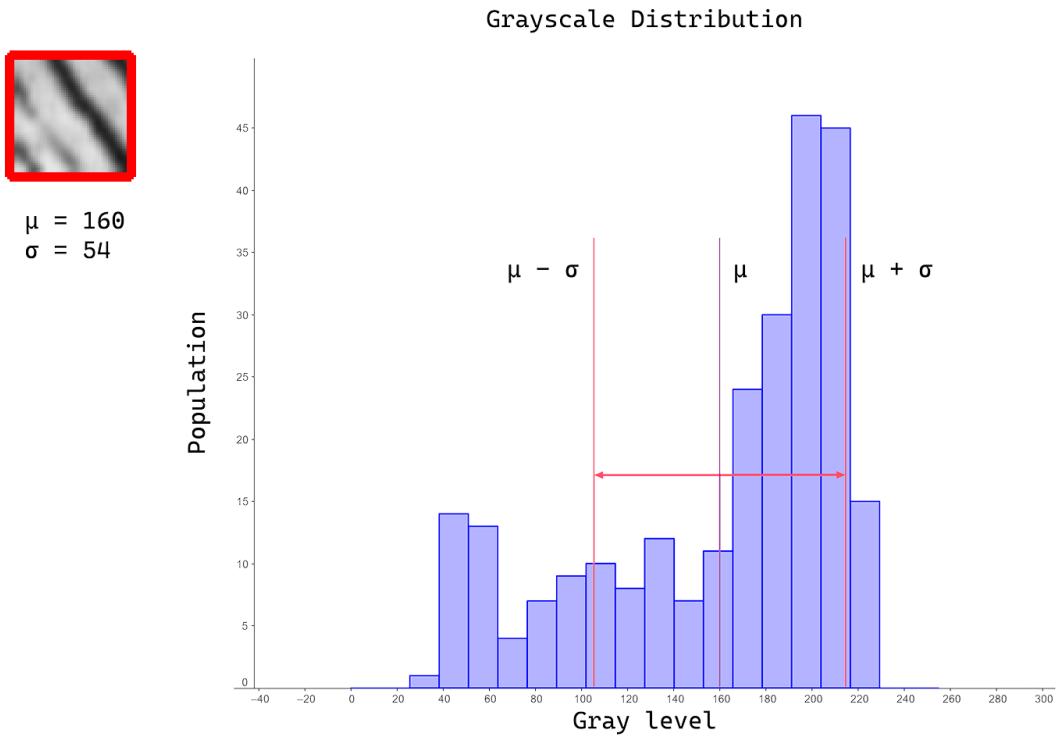


Figure 3: Grayscale Distribution of a good block

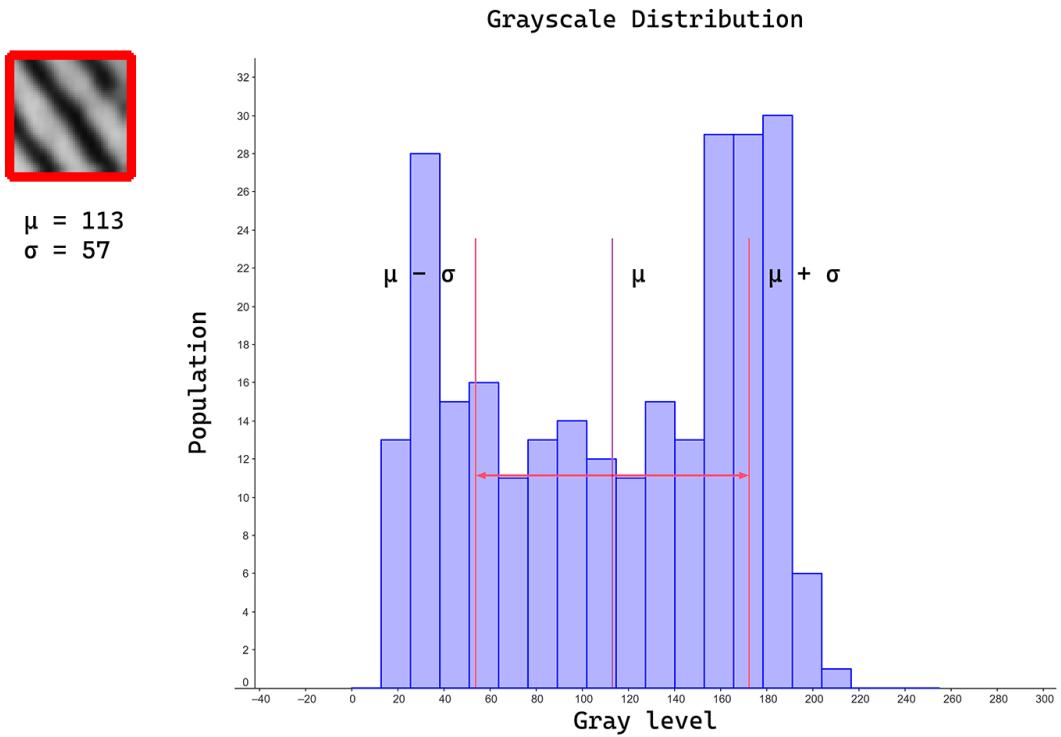


Figure 4: Grayscale Distribution of another good block

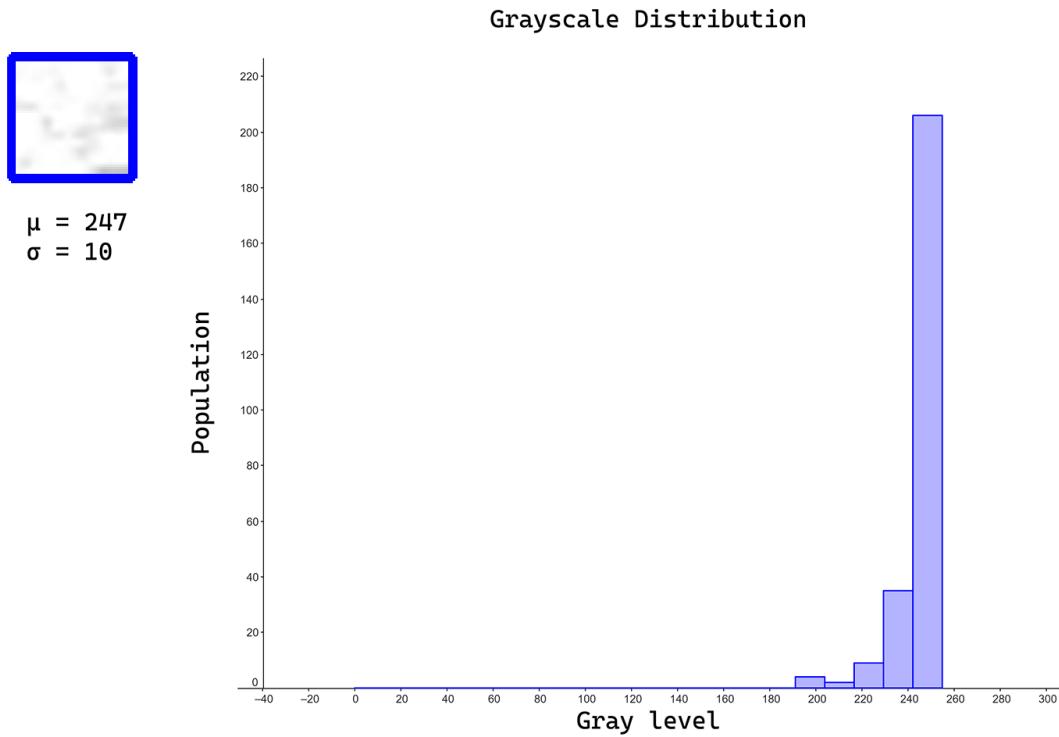


Figure 5: Grayscale Distribution of a bad block

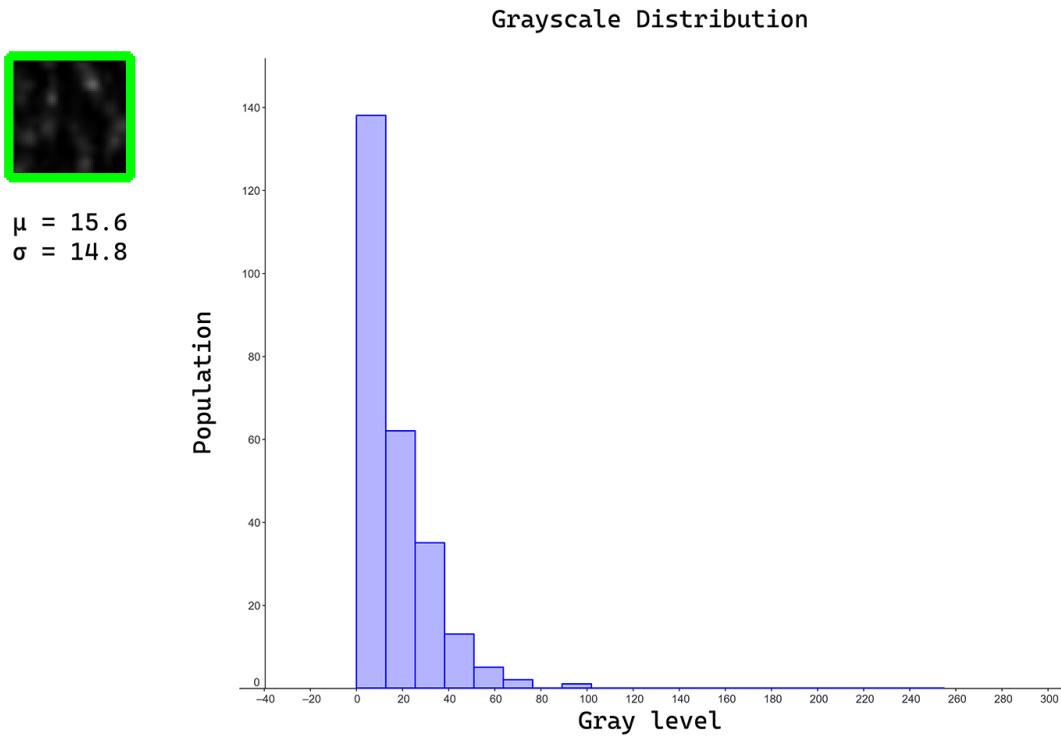


Figure 6: Grayscale Distribution of a background block



Figure 7: Masking blocks with $\text{std} < 0.2$

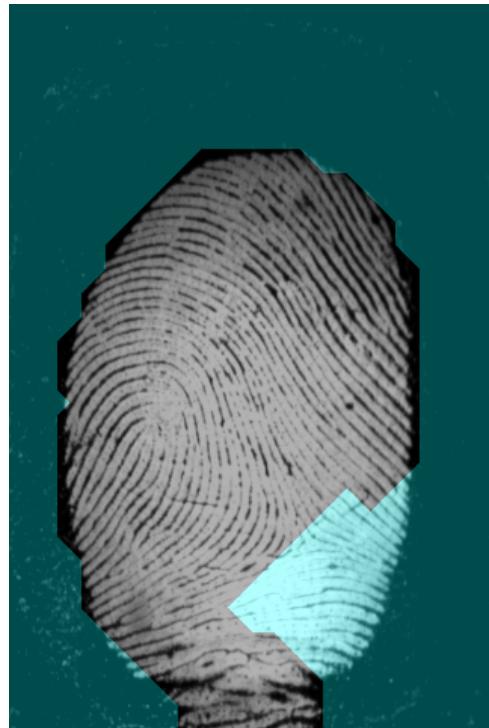


Figure 8: The smoothed mask

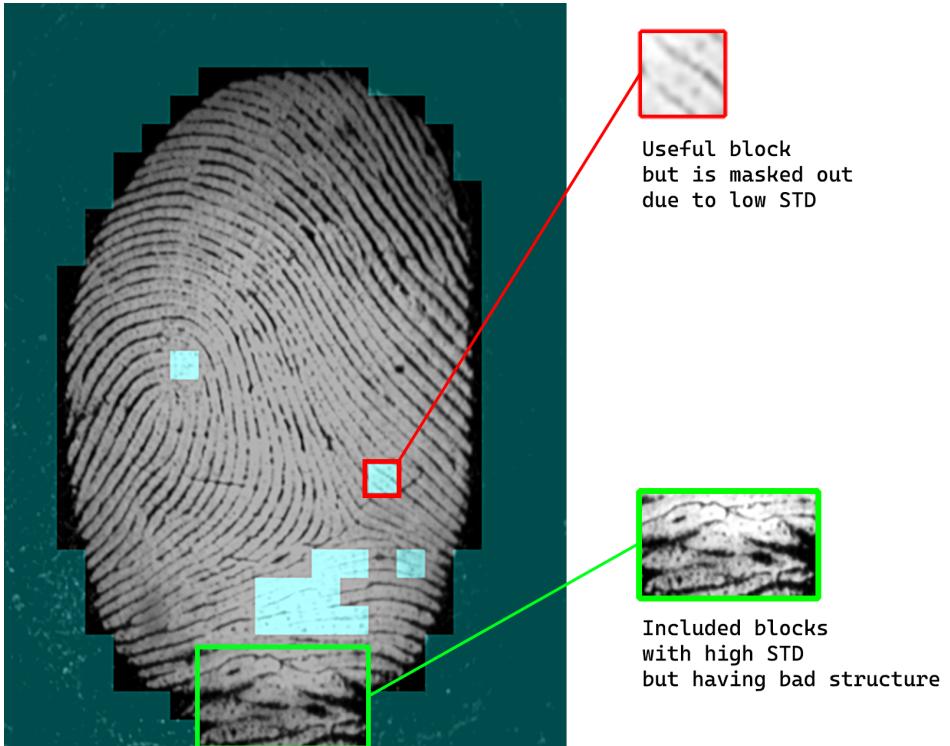
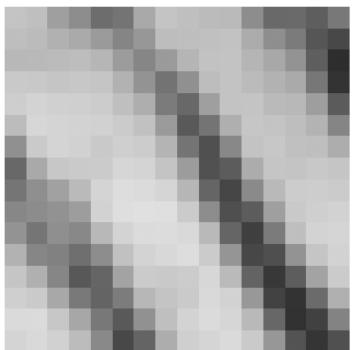


Figure 9: STD masking fails

Part 2: Orientation Certainty

$$gx = I * \begin{bmatrix} 1 & 0 & 1 \\ 2 & 0 & 2 \\ 1 & 0 & 1 \end{bmatrix}$$

$$gy = I * \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$



size: 16*16

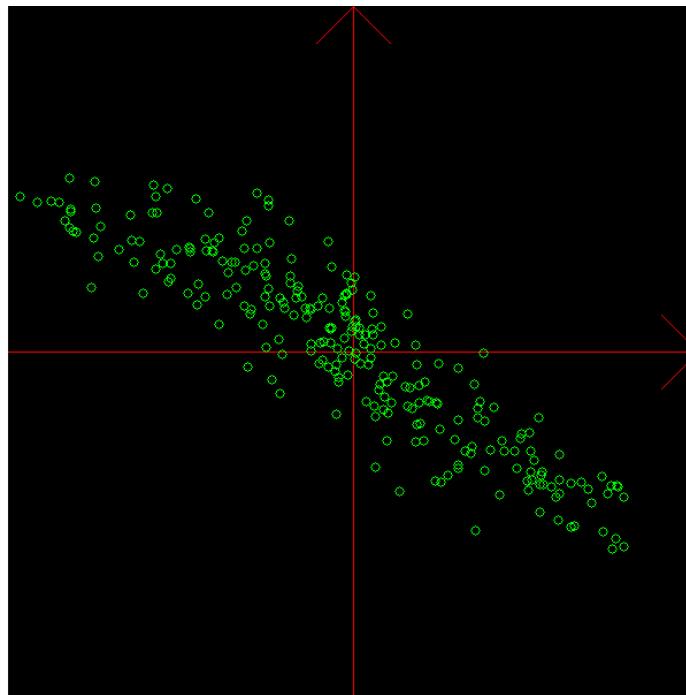
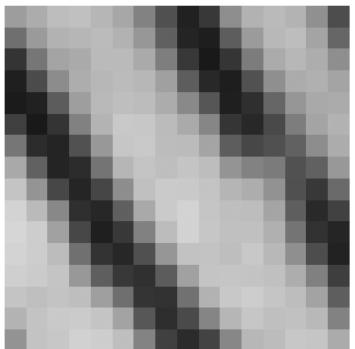


Figure 10: Relation between gx (horizontal axis) and gy (vertical axis) on a good block (1)



size: 16*16

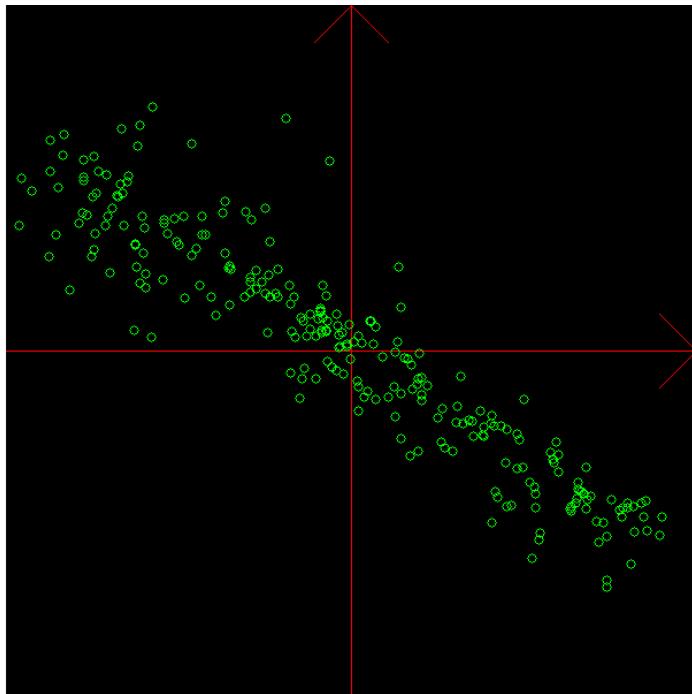


Figure 11: Relation between gx (horizontal axis) and gy (vertical axis) on a good block (2)



size: 16*16

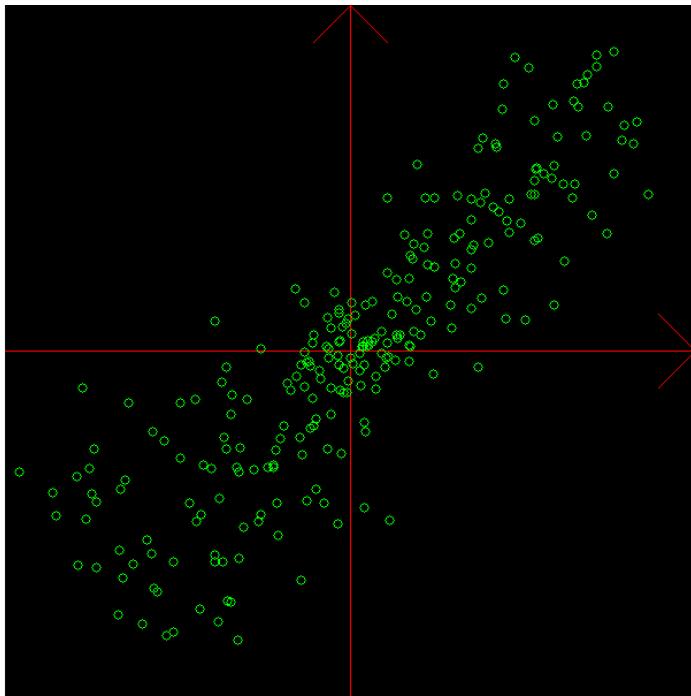


Figure 12: Relation between gx (horizontal axis) and gy (vertical axis) on a good block (3)



size: 16*16

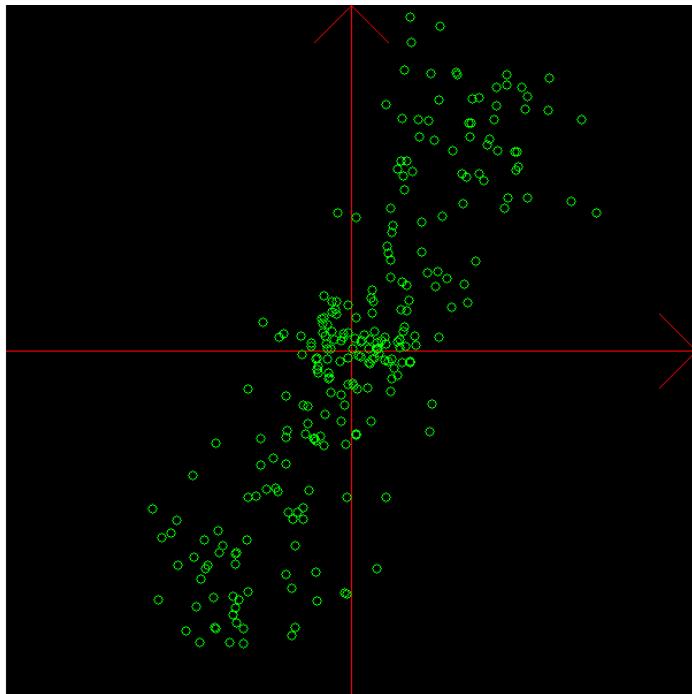
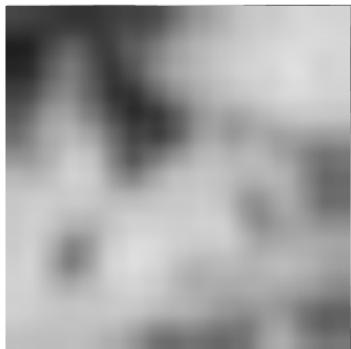


Figure 13: Relation between gx (horizontal axis) and gy (vertical axis) on a good block (4)



size: 16*16

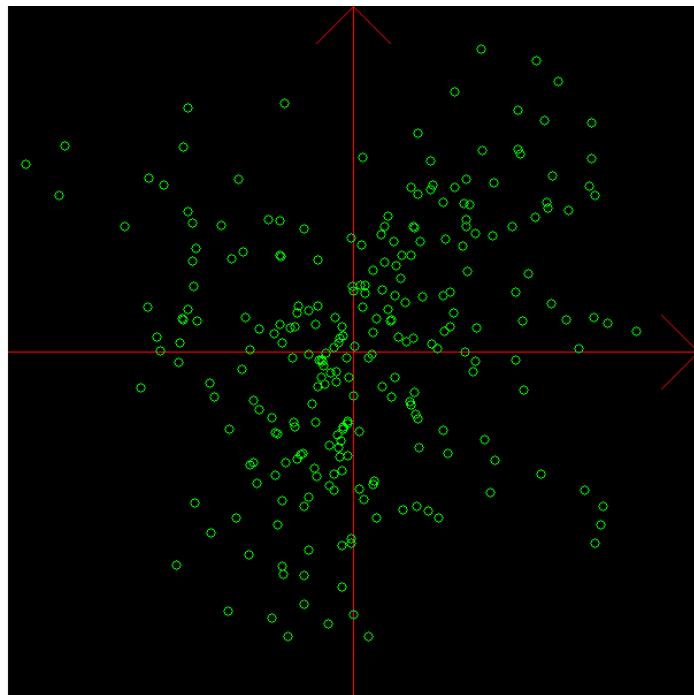


Figure 14: Relation between gx (horizontal axis) and gy (vertical axis) on a bad block (1)

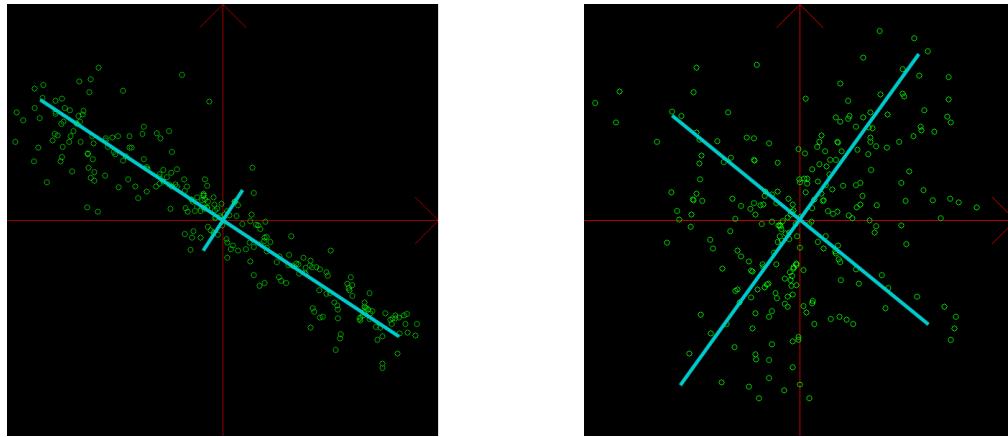


Figure 15: The direction along which the dataset has minimum/maximum variance
(the length of each line indicates the variance along respective direction)

Find unit vectors that minimize/maximize variance

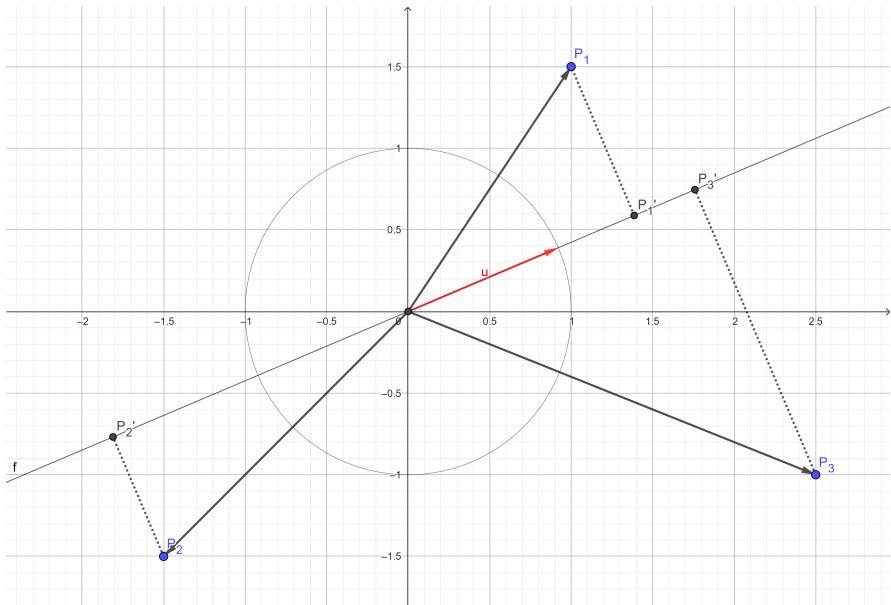


Figure 16: Projections of some points onto a unit vector

If \vec{v} is a unit vector:

$$dist(O, P'_i) = \vec{P}_i \cdot \vec{v} = \sum_{d=1}^D P_{id} v_d$$

Variance of projections:

$$V = \frac{1}{N} \times \sum_{i=1}^N \sum_{d=1}^D (p_{id} v_d - \mu)^2$$

By performing a geometric transformation such that

$$\mu_x = \mu_y = 0$$

Variable μ of the equation above then has the value of 0

Thus, the formula to calculate the variance of projections is simplified to:

$$V = \frac{1}{N} \times \sum_{i=1}^N \sum_{d=1}^D (p_{id} v_d)^2$$

The goal is to find a vector \vec{v} with the length of 1 unit
such that V is maximized.

And thus, I add a Lagrange multiplier λ to the equation:

$$V = \frac{1}{N} \times \sum_{i=1}^N \sum_{d=1}^D (p_{id} v_d)^2 - \lambda \left(\sum_{d=1}^D v_d^2 - 1 \right)$$

To find local min/max of V , I derive the equation into:

$$\frac{\delta V}{\delta v_a} = \frac{2}{N} \times \sum_{i=1}^N \left(p_{ia} \sum_{d=1}^D (p_{id} v_d) \right) - 2\lambda v_a$$

$$\text{At } \frac{\delta V}{\delta v_a} = 0:$$

$$\begin{aligned} & \frac{2}{N} \sum_{i=1}^N \left(p_{ia} \sum_{d=1}^D (p_{id} v_d) \right) = 2\lambda v_a \\ \Leftrightarrow & \sum_{d=1}^D v_d \left(\frac{2}{N} \sum_{i=1}^N p_{ia} p_{id} \right) = 2\lambda v_a \\ \Leftrightarrow & \sum_{d=1}^D v_d (2\text{cov}(p_a, p_d)) = 2\lambda v_a \end{aligned}$$

Since the image is two-dimensional, \vec{v} has 2 components v_x and v_y
The equation above is simplified into:

$$\begin{aligned} & \Leftrightarrow \begin{cases} v_x \text{cov}(p_x, p_x) + v_y \text{cov}(p_x, p_y) = \lambda v_x \\ v_x \text{cov}(p_y, p_x) + v_y \text{cov}(p_y, p_y) = \lambda v_y \end{cases} \\ & \Leftrightarrow \begin{bmatrix} \text{var}(p_x) & \text{cov}(p_x, p_y) \\ \text{cov}(p_y, p_x) & \text{var}(p_y) \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix} = \lambda \begin{bmatrix} v_x \\ v_y \end{bmatrix} \Leftrightarrow A\vec{v} = \lambda\vec{v} \end{aligned}$$

The fact that A is a matrix and λ is a scalar implies that
 \vec{v} must be an eigenvector

Consequently, the eigenvectors are the directions along which
the dataset has minimum/maximum variance

$$\left(\frac{\delta V}{\delta v_a} = 0 \right)$$

Next, I need to find the variance magnitude in the direction
of those eigenvectors, which is equivalent to the minimum/maximum variance
of the dataset along a direction

Breaking down the aforementioned variance of projections equation, I have:

$$\begin{aligned}
 V &= \frac{1}{N} \times \sum_{i=1}^N \sum_{d=1}^D (p_{id} v_d)^2 \\
 &= \frac{1}{N} \times \sum_{i=1}^N \left(\sum_{a=1}^D p_{ia} v_a \right) \left(\sum_{d=1}^D p_{id} v_d \right) \\
 &= \sum_{a=1}^D \sum_{d=1}^D \left(\frac{1}{N} \sum_{i=1}^N p_{ia} p_{id} \right) v_d v_a \\
 &= \sum_{a=1}^D \left(\sum_{d=1}^D cov(p_a, p_d) v_d \right) v_a
 \end{aligned}$$

With \vec{v} is an eigenvector
and $\sum_{d=1}^D v_d (2cov(p_a, p_d)) = 2\lambda v_a$ having been proven above,
the equation is then simplified into:

$$\begin{aligned}
V &= \sum_{a=1}^D (\lambda v_a) v_a \\
&= \lambda ||v||^2 = \lambda
\end{aligned}$$

Thus, the eigenvalues λ equal to the variance along the direction pointed by their corresponding eigenvectors

Consequently, the largest eigenvalue λ_1 is the maximum spread of the dataset along a direction.

While the smallest eigenvalue λ_2 indicate the minimum one.

By using these proofs and calculating λ_1 and λ_2 , I can define whether a block has a clear trend in orientation. The clearer the trend is, the larger λ_1 is compared to λ_2 . Otherwise, $\lambda_1 - \lambda_2$ approaches 0.

The covariance matrix of 2 gradient images gx and gy is calculated as:

$$A = \begin{bmatrix} var(dx) & cov(dx, dy) \\ cov(dy, dx) & var(dy) \end{bmatrix} = \begin{bmatrix} a & c \\ c & b \end{bmatrix}$$

We then find the eigenvalues λ_1 and λ_2 of A , which indicate the maximum and minimum variance of the dataset while being projected onto 1 line

$$\begin{aligned} A\vec{v} &= \lambda\vec{v} \\ \Leftrightarrow A\vec{v} &= \left(\lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right)\vec{v} \\ \Leftrightarrow A\vec{v} - \left(\lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right)\vec{v} &= \vec{0} \\ \Leftrightarrow \left(A - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right)\vec{v} &= \vec{0} \\ \Leftrightarrow \begin{bmatrix} a - \lambda & c \\ c & b - \lambda \end{bmatrix}\vec{v} &= \vec{0} \end{aligned}$$

Because \vec{v} is non-zero vector:

$$\begin{aligned} \det \begin{pmatrix} a - \lambda & c \\ c & b - \lambda \end{pmatrix} &= 0 \\ \Leftrightarrow (a - \lambda)(b - \lambda) - c^2 &= 0 \\ \Leftrightarrow \begin{cases} \lambda_1 = \frac{1}{2} \left((a + b) + \sqrt{(a - b)^2 + 4c^2} \right) \\ \lambda_2 = \frac{1}{2} \left((a + b) - \sqrt{(a - b)^2 + 4c^2} \right) \end{cases} \end{aligned}$$

Thus, λ_1 is the maximum variance of the dataset along one direction
while λ_2 is the minimum.

The orientation certainty level (*ocl*) of a block
can be then calculated as follow:

$$ocl = \frac{\lambda_2}{\lambda_1} \quad (0 \leq ocl \leq 1)$$

With low (high) ocl values, the local structure and orientation
of ridges and valleys are very regular (irregular), and therefore
the block has good (bad) quality.



Figure 17: Masking blocks using STD

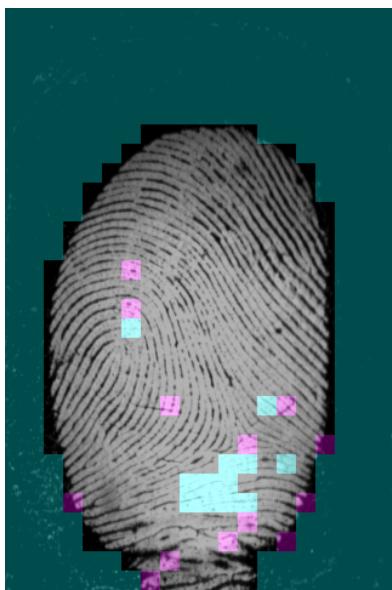


Figure 18: Combining the STD mask with OCL mask

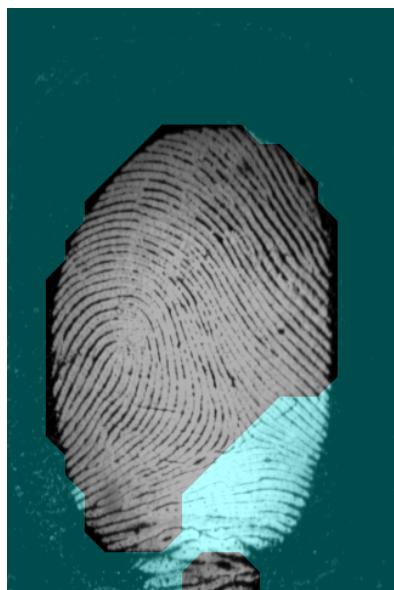


Figure 19: Smoothing the combined mask



Using STD and OCL to mask bad blocks out

(Left): Original Image (Middle): Masked image (Right): Skeleton from masked image



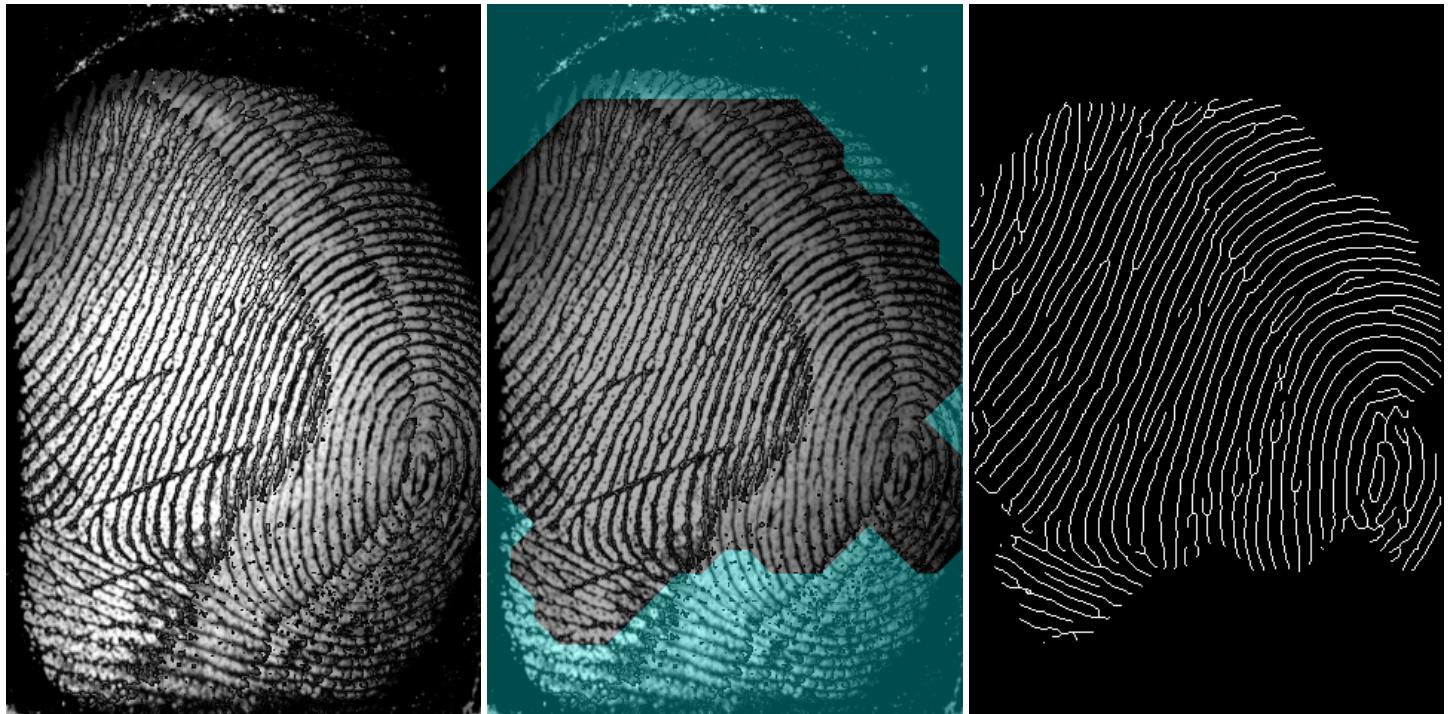
Using STD and OCL to mask bad blocks out

(Left): Original Image (Middle): Masked image (Right): Skeleton from masked image



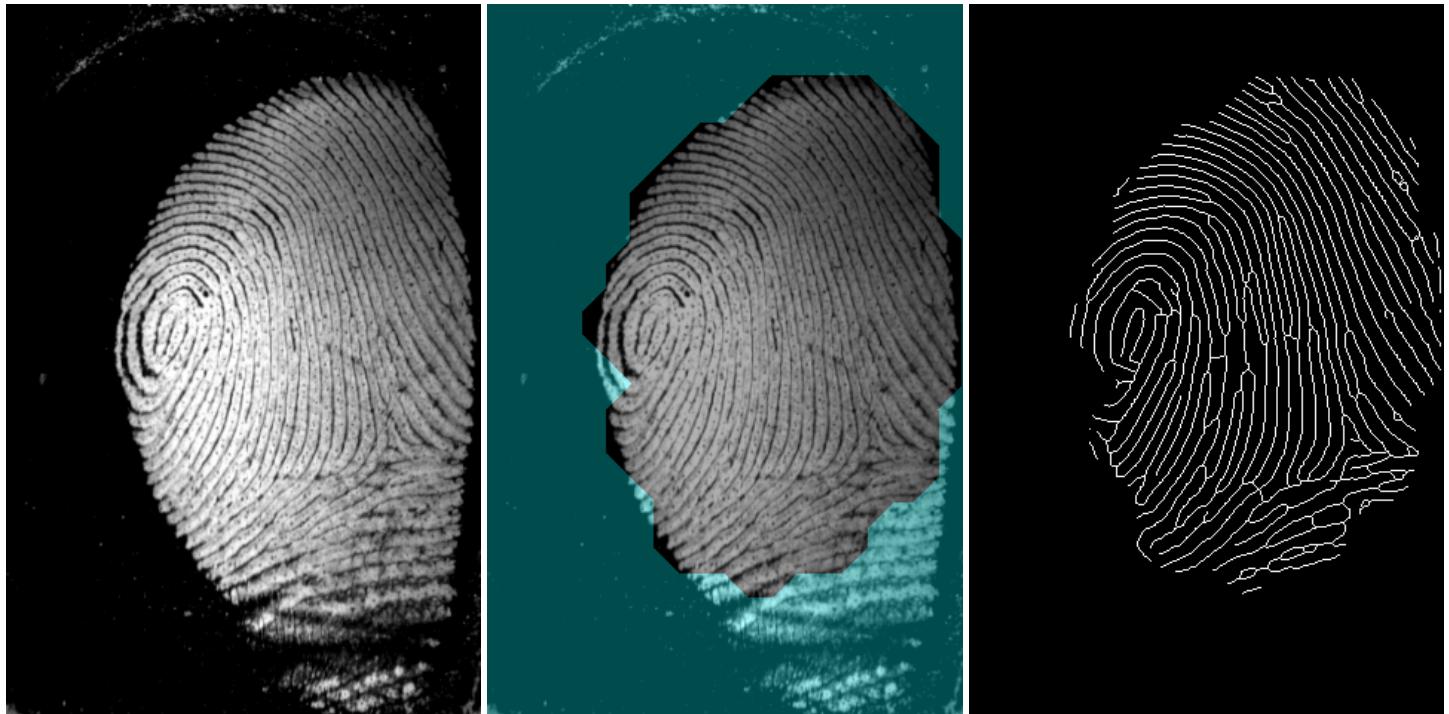
Using STD and OCL to mask bad blocks out

(Left): Original Image (Middle): Masked image (Right): Skeleton from masked image



Using STD and OCL to mask bad blocks out

(Left): Original Image (Middle): Masked image (Right): Skeleton from masked image



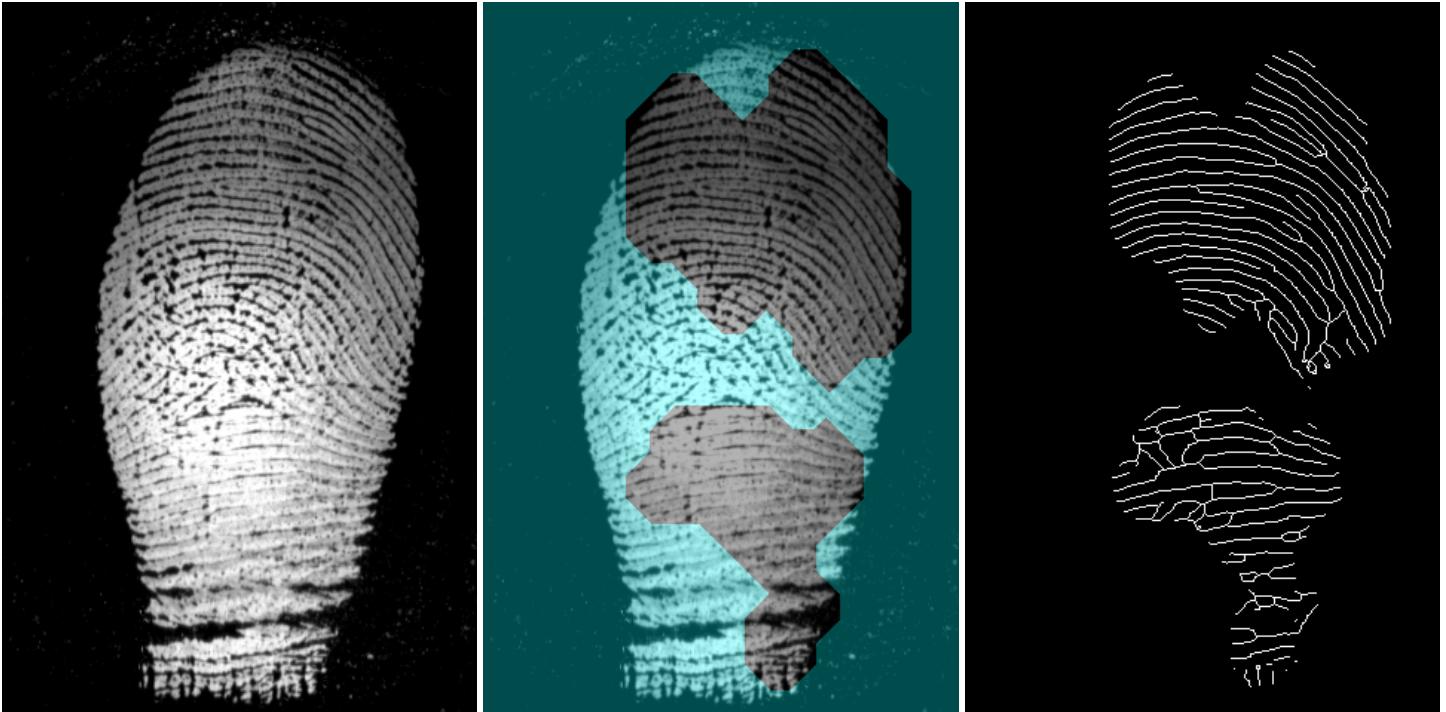
Using STD and OCL to mask redundant parts out

(Left): Original Image (Middle): Masked image (Right): Skeleton from masked image



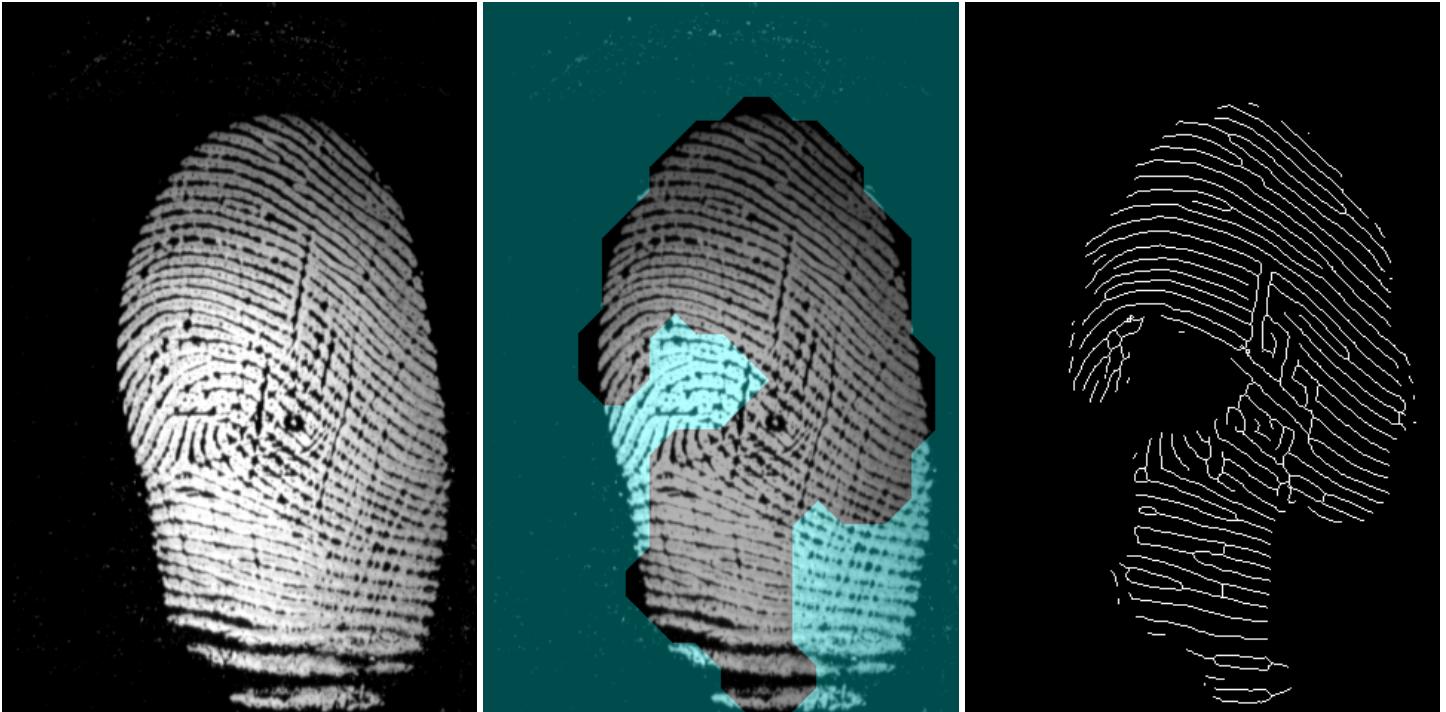
Using STD and OCL to mask redundant parts out

(Left): Original Image (Middle): Masked image (Right): Skeleton from masked image



Using STD and OCL gives clear hint of a bad image

(Left): A Bad Image (Middle): Mask of a Bad Image (Right): Skeleton from masked image



Using STD and OCL gives clear hint of a bad image

(Left): A Bad Image (Middle): Mask of a Bad Image (Right): Skeleton from masked image