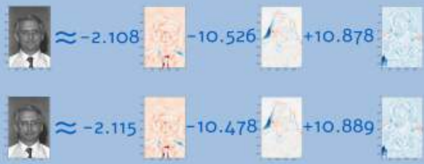
Face Recognition By Independent Component Analysis

HIT
מכון טכנולוגי חולון
Holon Institute of Technology

Student: Oria Gruber
Supervisor: Dr. Yirmeyahu Kaminski

The Goal:

The project's goal is to propose a way to implement facial recognition using a form of Blind Source Separation called: Independent component analysis (ICA).



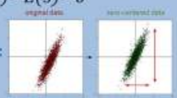
As shown, the representation of both images with respect to the ICA basis is similar - A match was found.

Overview:
We have our sample vactor x , ICA wil find a matrix A and a source vector s such that $x=As$ and s are statistically independent.

Step One: Preprocessing
The first step in ICA is the preprocessing step. This step is divided to 2 stages:
Centering and Whitening.
Preprocessing is done to speed up the convergence of the ICA iterative process and increase numerical precision.

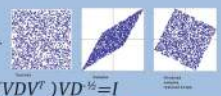
Centering is the process of removing the mean from our samples: $\tilde{x}=x-E(x)=As-E(As)=A(s-E(s))=A\tilde{s}$
This process yields a new ICA problem which is easier, since $E(\tilde{x})=E(\tilde{s})=0$

In effect, this is equivalent to moving our samples to the origin:



Another result of centering is that we can now assume that $Cov(\tilde{s})=I$, this is because $Cov(\tilde{s})=E((\tilde{s}-E(\tilde{s}))\,(\tilde{s}-E(\tilde{s}))^T)=E((\tilde{s}-0)\,(\tilde{s}-0)^T)=E(\tilde{s}\tilde{s}^T)=I$

Whitening is the process of restoring the shape of the samples to the shape of the original sources.



Let $Cov(x)=VDV^T$. If we define $Q:=D^{-1/2}V^T$, we observe $Cov(Qx)=E(Qxx^TQ^T)=QCov(x)\,Q^T=D^{-1/2}\,V^T\,(VDV^T)VD^{-1/2}=I$
The result is that $Qx=QAs$ is a simpler than our original problem since $Cov(Qx)=I$ and QA is orthogonal, thus Qx is merely rotated s.

Step Two: Source Separation
After the preprocessing phase, our samples x are uncorrelated, like our source vector s, but unlike s, they are not independent.
Our goal is to find an orthogonal matrix W such that $y=Wx$ are independent. This matrix W will rotate x back to the original s.

The mutual information of y is $MI(y)=\int_{\Omega}P(y_1,...,y_n)\log\left(\frac{P(y_1,...,y_n)}{P(y_1)...P(y_n)}\right)d\vec{y}$, this function is not negative and is 0 if and only if y are independent.

Thus we want to find W such that $MI(y)=MI(Wx)$ is minimal. From Central Limit Theorem this can be shown to be equivalent to maximizing the kurtosis:
$k(w_ix)=k(y_i)=E(y_i^4)-3$. Using Newton's method we can find w_1 that maximizes $k(w_1x)$ under the constraint $|w_1|=1$.
In a similar fashion we can find Wi if we add the orthogonality constraints, and we fill our W matrix row by row in this manner. allowing us to find $s=Wx$.

Step Three: Facial Recognition
Let $x_1,x_2,...,x_n$ be our images in row vector form. We insert them to matrix X such that x_1 is the first row and so on.
ICA will find a source matrix S such that $s_1,s_2,...,s_n$ are a basis that spans the face-space of our matrix X, and a coefficient matrix A such that $X=AS$.
Thus, the representation of x_i with respect to basis S is A_i.
Let b_i,b_j be the representation of x_i,x_j respectively in basis S.
We say that there is a match between x_i and x_j (meaning the person in image i is the same as the person in image j)
if the cosine norm $d^2(b_i,b_j)=|b_i|+|b_j|-2|b_i||b_j|\cos\theta$ is sufficiently small.
Since they are normalized vectors, this is equivalent to saying the cosine of the angle between b_i and b_j is larger than 1-ε.

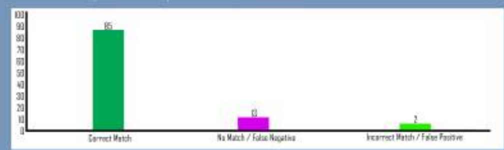
Difficulties:
Most of the difficulties I had during this project were technical difficulties of the implementation in Python.

Implementing ICA and performing facial recognition required a number of packages that were not available in Windows OS, as such, I was required to install Ubuntu and work with a language and environment which I've never worked before.

The project also demanded a very high level of knowledge in the fields of Linear Algebra, Probability and Numerical Methods, and I was required to broaden my horizons beyond what was taught to me in our classes.

Python Implementation:

Statistical Model:
100 different expression experiments with $\varepsilon=0.5$



Conclusions:
ICA works very well when recognition is performed on images that are taken from the same angle. Infact, it performs roughly 25% better than PCA or similar algorithms.
When the images are taken from a different angle, ICA performs poorly and only managed to correctly recognize the person 15 out of 100 experiments.

As such, it is a very powerful tool, but only to be used in certain situations.

Enhancements:
ICA performance could theoretically be enhanced if we investigate a Machine Learning approach.
I did not implement a learner due to time and resources constraints, however, an optimal route to take would be such that every time the computer recognizes a person, it adds the new picture to the database of face images.
That way, when we try to recognize the person again, the computer will have a better chance of recognizing him, it will have more reference points, it will learn and improve the more times we use it.

This will result in less cases of false negatives and false positives.