# 3D FACIAL FEATURE MODELING WITH AAM


**MATTEO SORCI**
MATTEO.SORCI@EPFL.CH

**TIM LLEWELYNN**
TIM.LLEWELLYNN@NVISO.CH

**HADRIEN COPPONEX**
HADRIEN.COPPONEX@EPFL.CH


## PROBLEM

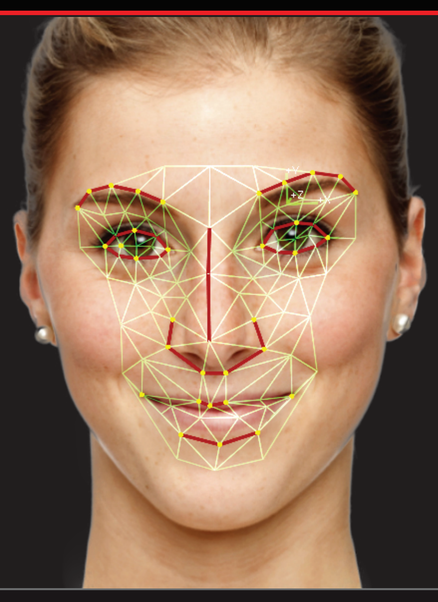
Active Appearance Models (AAM) are powerful set of tools for modeling and matching objects under shape deformations and texture variations. It learns characteristics of objects by building a compact statistical model from applying Principal Component Analysis (PCA) to a set of labeled data. AAM has been widely applied in the fields of computer vision, due to its flexible and simple framework, however it still cannot satisfy the requirement of real-time situations. To alleviate this problem, computational complexity of either training or fitting procedures should be considered, which involves texture representation, optimization algorithm, and model training. We address this drawback by running the entire AAM algorithm on the GPU and exploiting a hybrid CPU / GPU block processing architecture. The algorithm is tested using a dedicated 143 point face model with a range of applications including :

- Automotive Driver Monitoring
- Markerless MOCAP and Emotion Recognition
- Non-Verbal Behavior Analysis in Video Streams
- Interactive Gaming and Entertainment
- Security and Face Recognition Systems
- Retail Kiosks and Market Research

## ACTIVE APPEARANCE MODELS

The goal in AAM is to fit a shape model to an image using the appearance information from the image. The statistical shape model is learned from the shape vectors x = (x1,y1,x2,y2, ...,xn,yn) defining the x and y image coordinates of n landmarks. The shape vectors can then be summarized by PCA, yielding a mean and basis (shape) vectors. Any shape can then be represented as a linear combination of this mean and basis shapes.

Statistical texture model is defined over shape-free representations which are obtained after affine warping the images to the mean shape. After normalizing their means and standard deviations, the texture can be represented as a linear combination of the mean appearance and the basis appearances obtained with PCA. Fitting a model to an image consists in finding the appearance parameters that minimise the difference between the target image and the one synthesised by the appearance model.



## HYBRID CPU / GPU SOLUTION

INPUT IMAGE OR VIDEO SEQUENCE

CPU

FACE DETECTION → PRE-PROCESSING → TEXTURE

GPU

TEXTURE

ACTIVE APPEARANCE MODEL

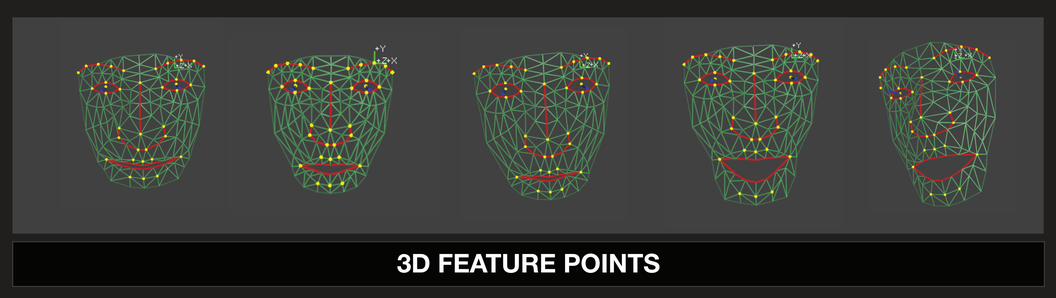
SHAPE
$x = (x_1, y_1, \ldots, x_n, y_n)^T$

PCA
$x = \bar{x} + P_s b_s$

$b_c = \begin{pmatrix} Wb_s \\ b_g \end{pmatrix} = Qc = \begin{pmatrix} Q_s \\ Q_g \end{pmatrix} c$

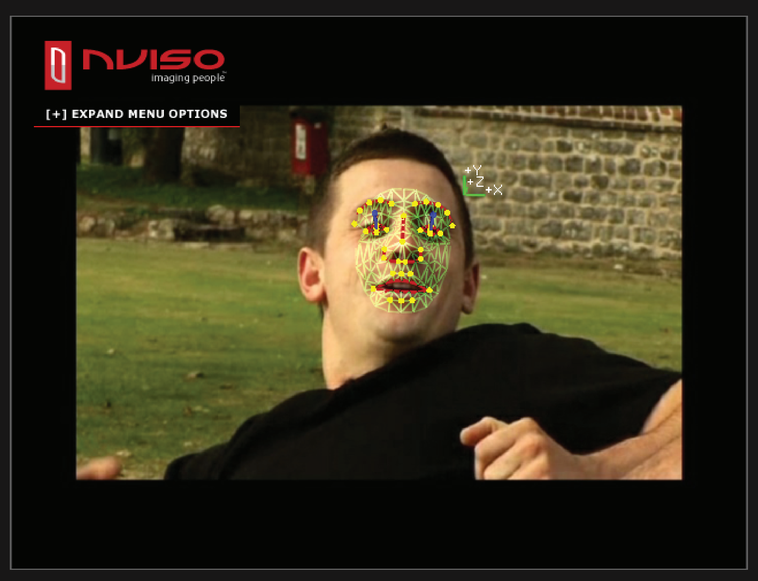
TEXTURE
$g = (g_1, \ldots, g_m)^T$

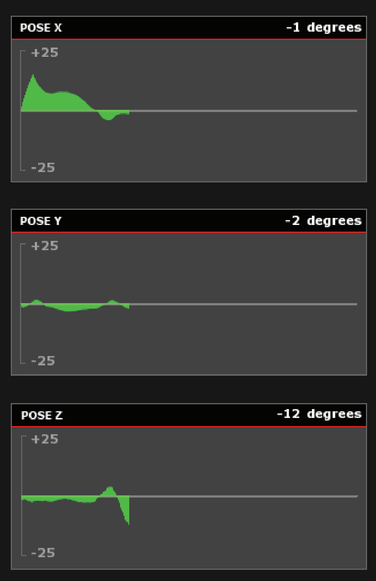
PCA
$g = \bar{g} + P_g b_g$

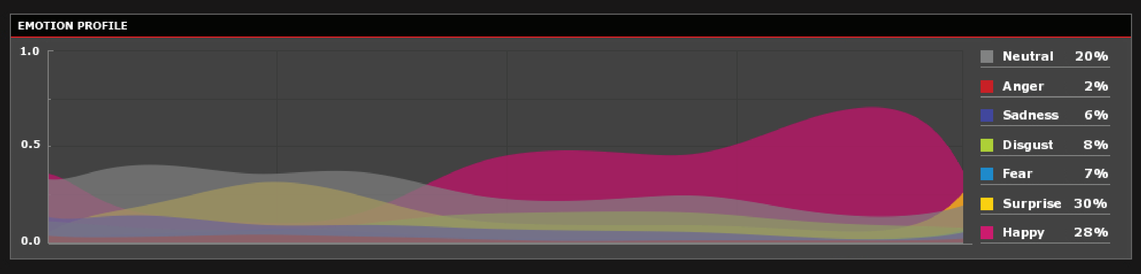


3D FEATURE POINTS

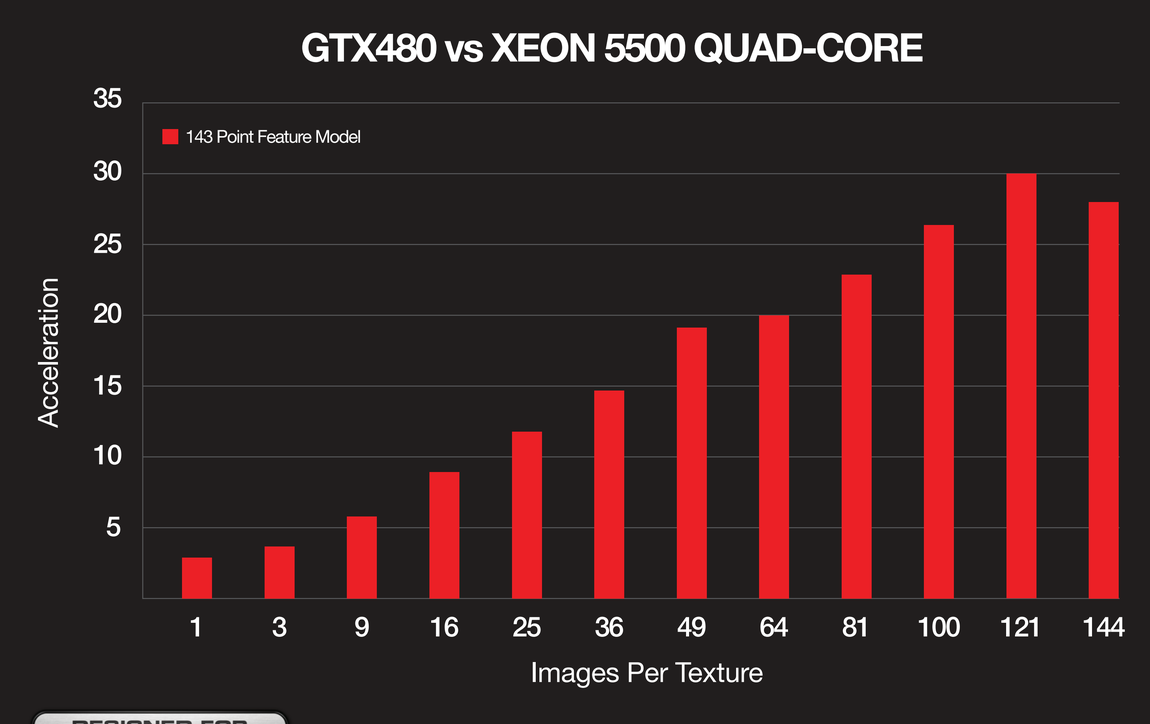
## REAL-TIME APPLICATION




## PERFORMANCE VS BLOCK SIZE




DESIGNED FOR NVIDIA CUDA

**29 x**

Hybrid GPU 1 sec

OVERALL TIME ON 339 IMAGE DATABASE

CPU XEON 5500 QUAD CORE 29 sec

nviso imaging people™

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE


CREDITS : The authors would like to thank the Swiss National Science Foundation who kindly funded part of this research project. The LTS5 signal processing laboratory of the EPFL for thier interest and support of this work. Special thanks goes to Mario Dzurila and Shervin Emami for help in preparing the user interface and facial feature visualization.