

Interpolating position

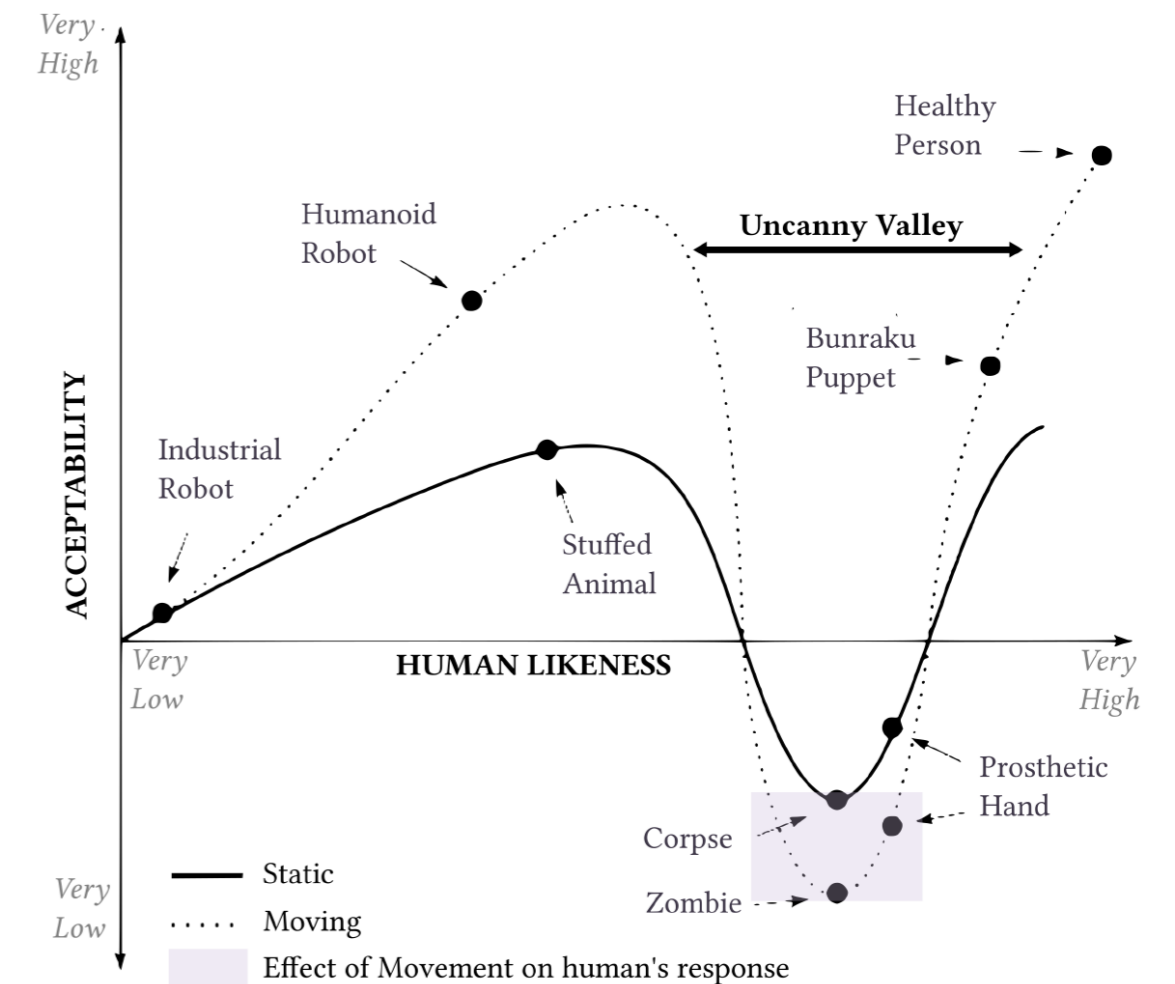
Facial Animation: Blend Shapes

Facial animation

Facial expressions are particularly hard to automatically model

- Lots of details (huge number of muscles)
- Human perception is very sensitive on face

Risk of *Uncanny Valley*



Usage of keyframes interpolation

Facial expression are categorized:



Interpolate every vertex between multiple meshes (key-poses)



Pose b_1

Interpolation $p(t)$

Pose b_2

$$p_i(t) = \omega_1(t)b_{i1} + \omega_2(t)b_{i2}$$

Blend Shapes: Multi-target blending

Interpolate between multiple key poses

Interesting for facial animation

ex. 30% happy, 50% suprised, 20% tired

- Per-vertex formulation $p_i(t) = \sum_k^{N_{poses}} \omega_k(t) b_{ik}$

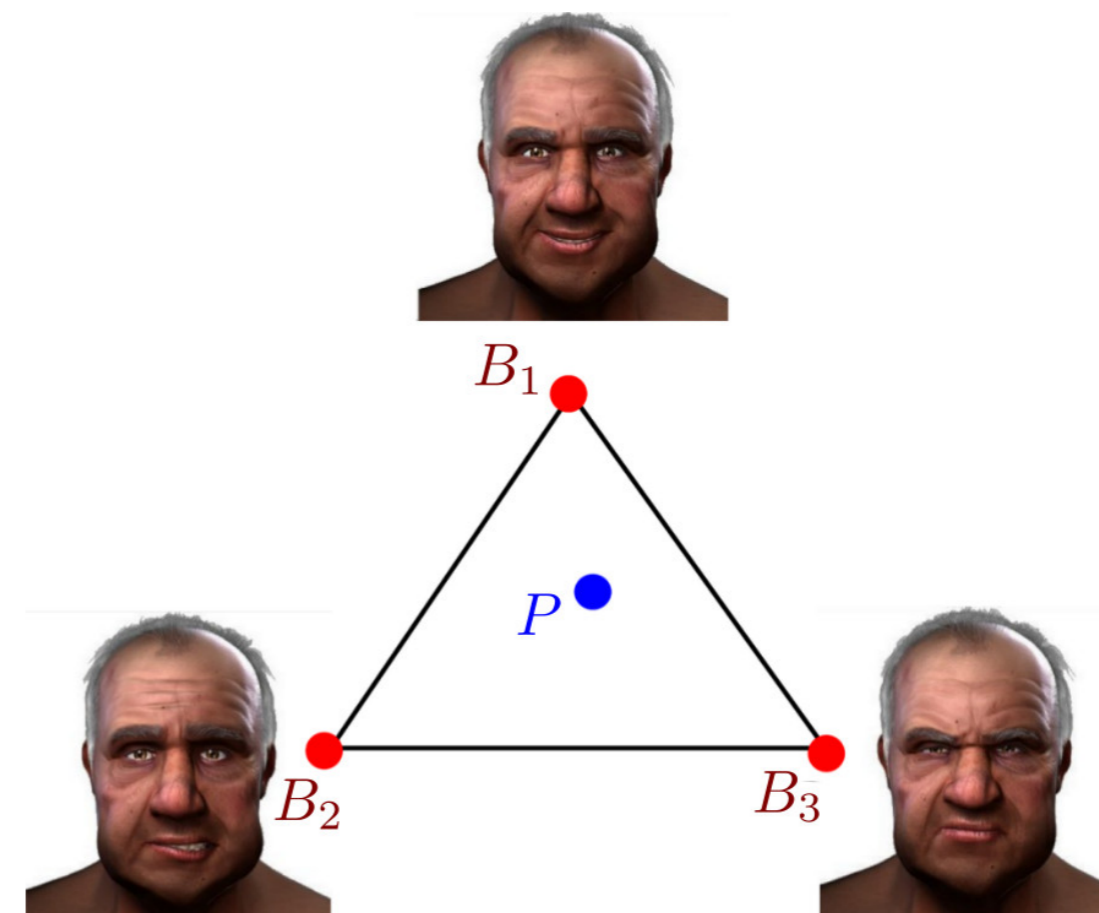
- p_i : i th deformed vertex coordinates

- b_{ik} : Initial i th vertex coordinates at pose k

- w_k : Interpolation weight for pose k .

- Matrix formulation $\mathbf{p}(t) = \mathbf{B} \omega(t)$

$$\begin{pmatrix} p_1(t) \\ p_2(t) \\ \vdots \\ p_N(t) \end{pmatrix} = \underbrace{\begin{pmatrix} b_{11} & \dots & b_{1N_{poses}} \\ b_{21} & \dots & b_{2N_{poses}} \\ \vdots & & \vdots \\ b_{N1} & \dots & b_{NN_{poses}} \end{pmatrix}}_{\text{all vertices from all poses}} \underbrace{\begin{pmatrix} \omega_1(t) \\ \vdots \\ \omega_{N_{poses}}(t) \end{pmatrix}}_{\text{weights per pose}}$$



Coordinate differences

Often simpler to add/remove some expression w/r neutral pose

Add smile, open more the left eye, etc.

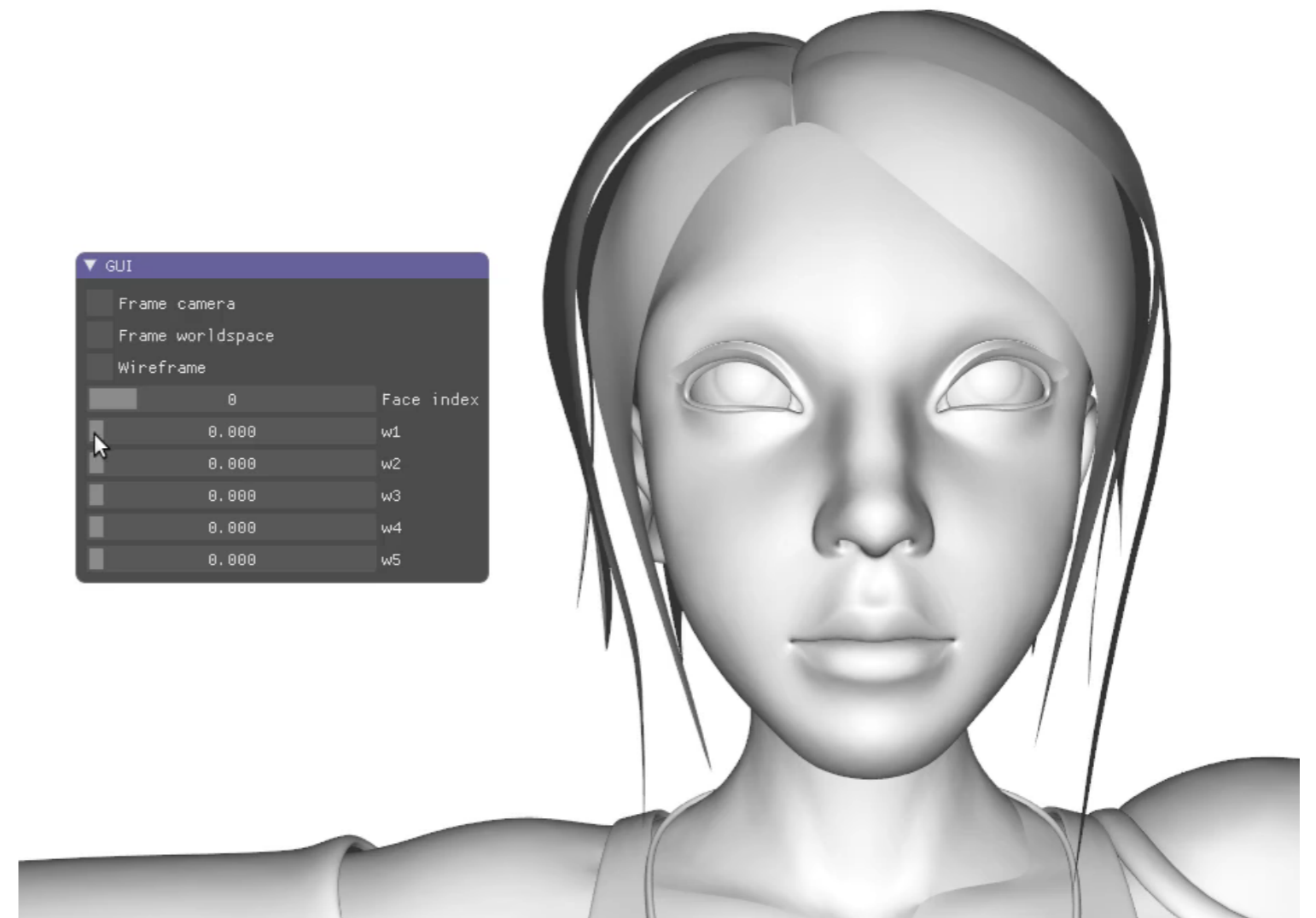
Idea:

- Set b^0 to be neutral pose
- Precompute *poses differences* $d_k = b_k - b^0$

- Compute
$$p_i(t) = b_i^0 + \sum_k^{N_{poses}} \omega_k(t) d_{ik}$$

In matrix formulation $\mathbf{p}(t) = \mathbf{b}^0 + \mathbf{D} \omega(t)$

D : matrix of vertex differences between poses.



Blend shapes - Summary

Standard for facial/complex shapes deformation

(+) Allow to add/subtract local expressions from pre-set poses
even extrapolate

(+) Allow expression transfert

Keep same D for difference faces b^0

Requires same connectivity

Requires non redundant/interfering poses

D is not an othogonal matrix

Different weights can give the same result

PCA can orthogonalize the matrix

but weights are less intuitive (not positive)

To go beyond:

[J.P. Lewis et al. *Practice and Theory of Blendshape Facial Models*. EG STAR 2014]

