

Layered Compositing of Facial Expression

Ken Perlin
Media Research Laboratory
Department of Computer Science
New York University

How does one make an embodied agent react with appropriate facial expression, without resorting to repetitive prebuilt animations? How does one mix and transition between facial expressions to visually represent shifting moods and attitudes? How can authors of these agents relate lower level facial movements to higher level moods and intentions? We introduce a computational engine which addresses these questions with a stratified approach. We first define a low level movement model having a discrete number of degrees of freedom. Animators can combine and layer these degrees of freedom to create elements of autonomous facial motion. Animators can then recursively build on this movement model to construct higher level models.

In this way, animators can synthesize successively higher levels of autonomous facial expressiveness. A key feature of our computational approach is that composited movements tend to blend and layer in natural ways in the run time system. As the animator builds at higher levels, the correct layering priorities are always maintained at lower supporting levels. We have used this approach to create emotionally expressive autonomous facial agents.



angry, daydreaming, disgusted, distrustful



fiendish, haughty, head back, scolding, sad



smiling, sneezing, surprised, suspicious

Building on our work in *Improvisational Animation* [Siggraph 96], we use a parallel layered approach.

Our authoring system allows its user to relate lower level facial movements to higher level moods and intentions by using a model inspired by optical compositing. We first allow the animator to abstract facial motion into a discrete number of degrees of freedom. The system does not impose a particular model at this stage, but rather allows the animator to define the degrees of freedom that he or she finds useful.

Given a set of degrees of freedom, we allow the animator to specify time varying linear combinations and overlays of these degrees of freedom. Each definition becomes a new, derived, degree of freedom. Collectively, these derived degrees of freedom create a new abstraction layer. We allow animators to recursively create successive abstractions, each built upon the degrees of freedom defined by previous ones.

More specifically, we internally represent each of the model's K degrees of freedom by a K dimensional basis vector, which has a value of 1 in some dimension j , and a value of 0 in all other dimensions. The state space for the face consists of all linear combinations of these basis vectors. We allow the animator to create time varying functions (coherent noise and splined curves) of these linear combinations, to create a simple vocabulary of "basis motions". We then provide a motion compositing engine, which allows the animator to specify a compositing structure for these basis motions. The engine gives motions varying "opacities" and does alpha blending, much as Photoshop composites layers of images. The animator can then encapsulate sets of time-varying weights for each of the basis motions in the above engine, and define each such set as a new motion basis. By doing this recursively, the animator can describe successively higher levels of an emotional vocabulary.

We have found that in this way, successively higher semantic levels of facial expressiveness can be effectively synthesized. In particular, we find that movements defined with this method tend to blend together and overlay in natural ways. As the animator works within higher abstractions, the system always maintains correct priorities at all lower supporting abstractions. The result is real-time interactive facial animation that can achieve a convincing degree of emotive expressiveness.

The eventual goal of this work is to give computer/human interfaces the ability to represent the subtleties we take for granted in face to face communication, so that they can function as agents for an emotional point of view. By automating the creation of facial expression in the run-time engine of an interactive agent, we enable such an agent to operate without the explicit intervention of a human operator.