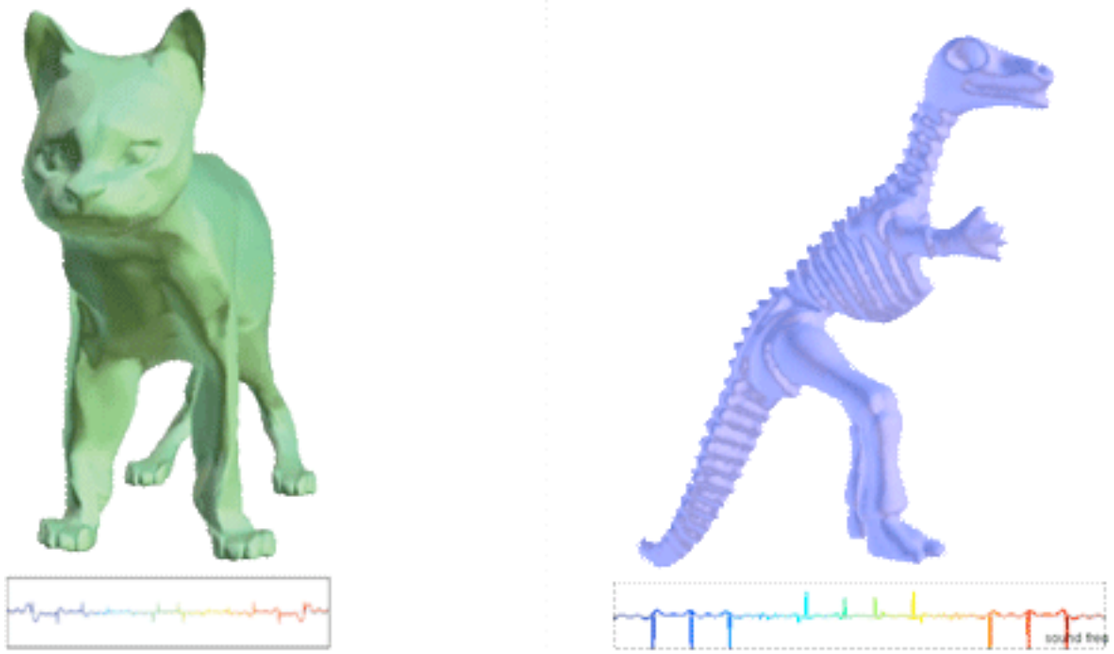


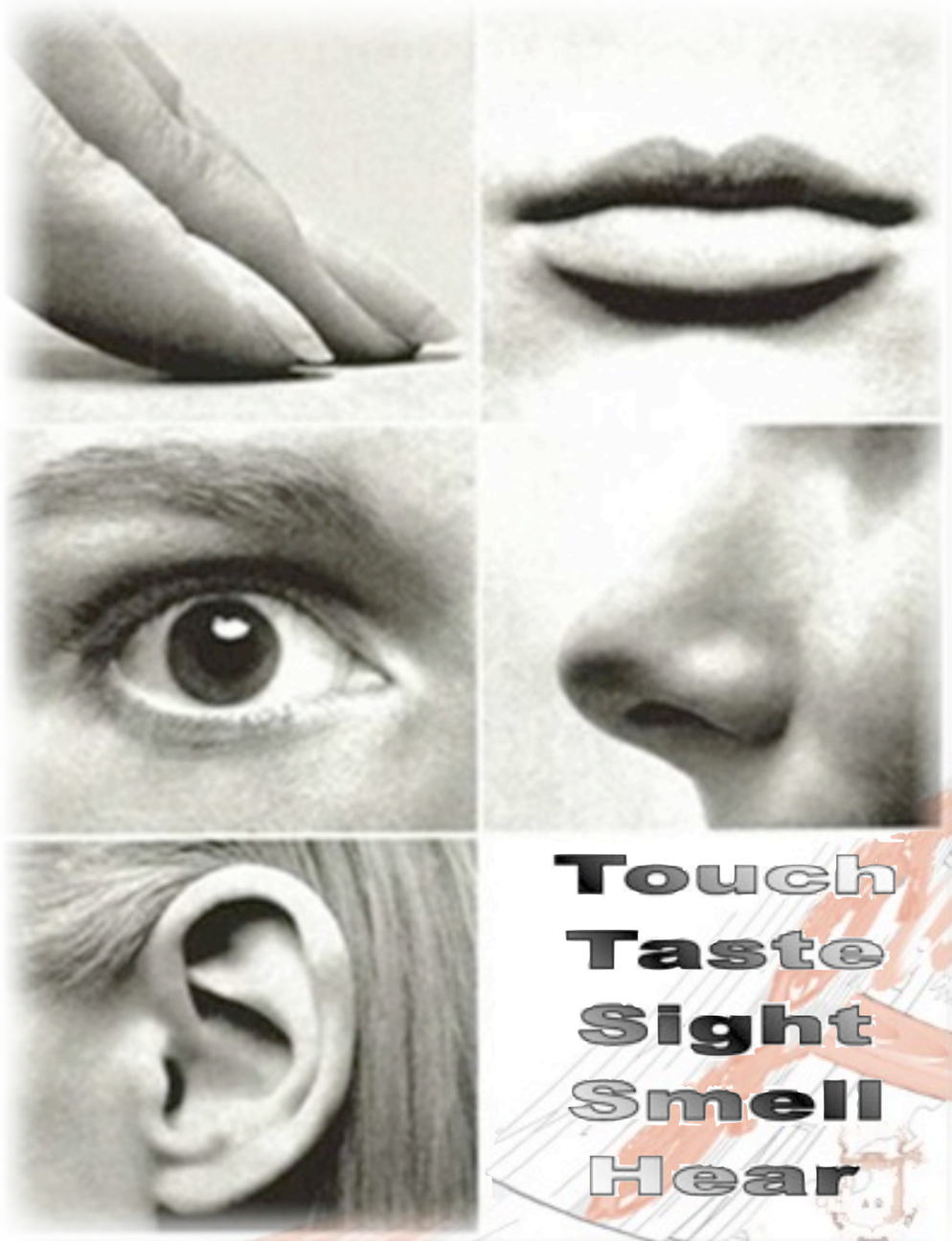
Tuning Manifold Harmonics Filters

Thomas Lewiner, Thales Vieira, Alex Bordignon,
Allyson Cabral, Clarissa Marques, João Paixão,
Lis Custódio, Marcos Lage, Maria Andrade,
Renata Nascimento, Scarlett de Botton,
Sinésio Pesco, Hélio Lopes, Vinícius Mello,
Adelailson Peixoto, Dimas Martinez

PUC-Rio de Janeiro, UFAL & UFBA - Brazil

Matmidia

Enhancing Music



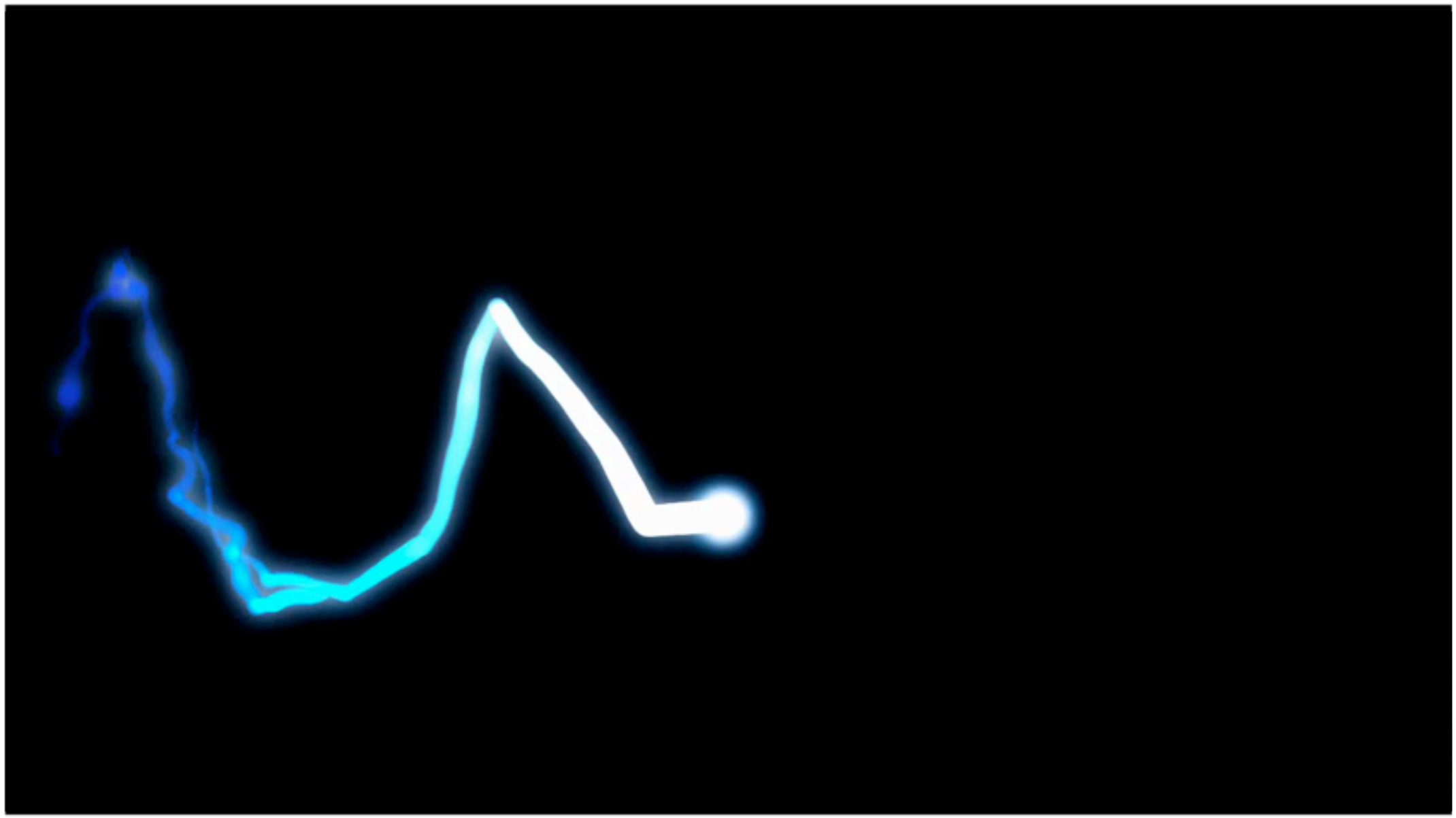
Music Clips / 3d Animation

2 Rios / Skank (Vetor Zero, Sibgrapi Video Festival 2003)

Laborious Synchronization

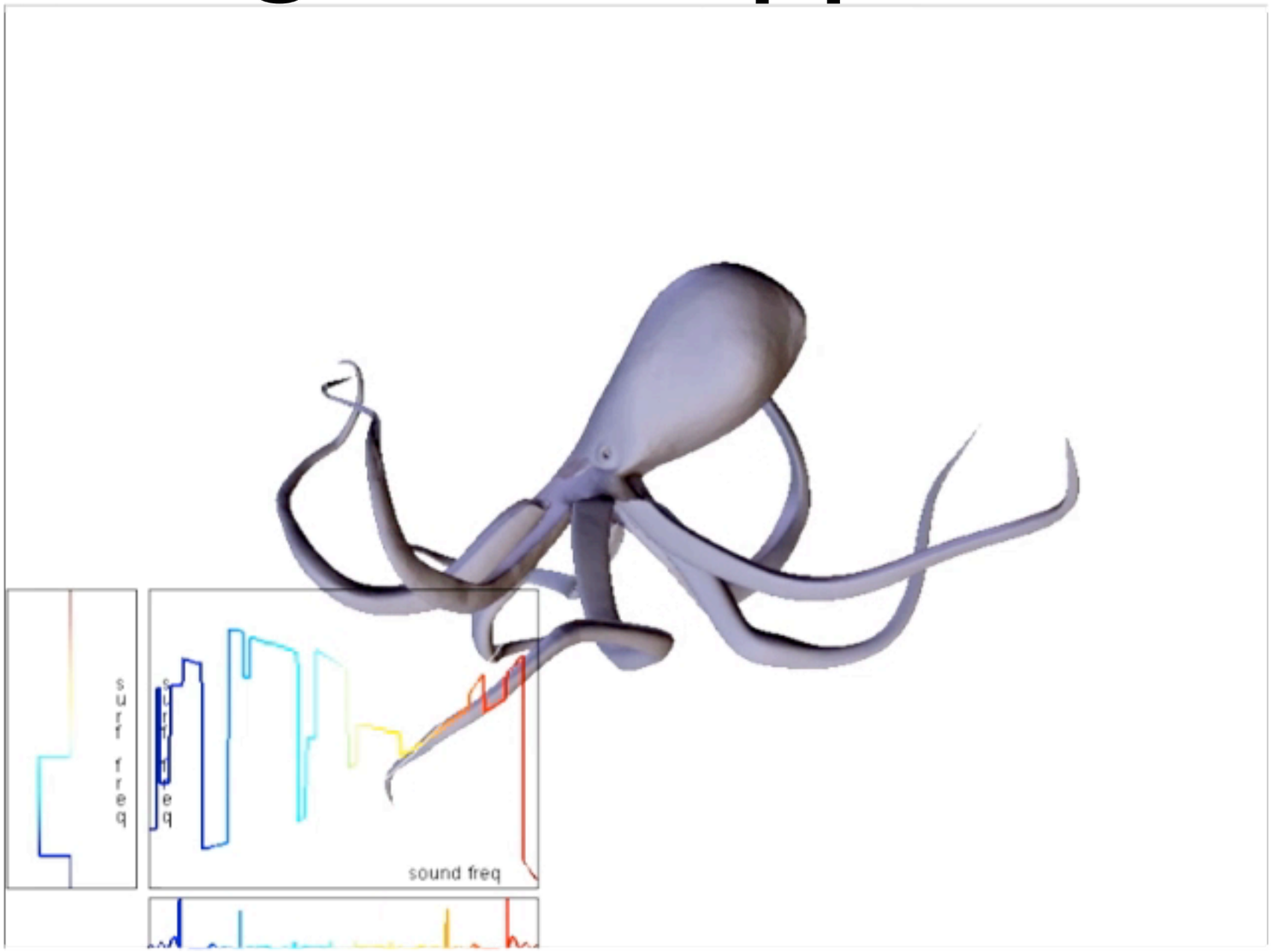
motion capture or manual synchronization

Synchronized Visualization



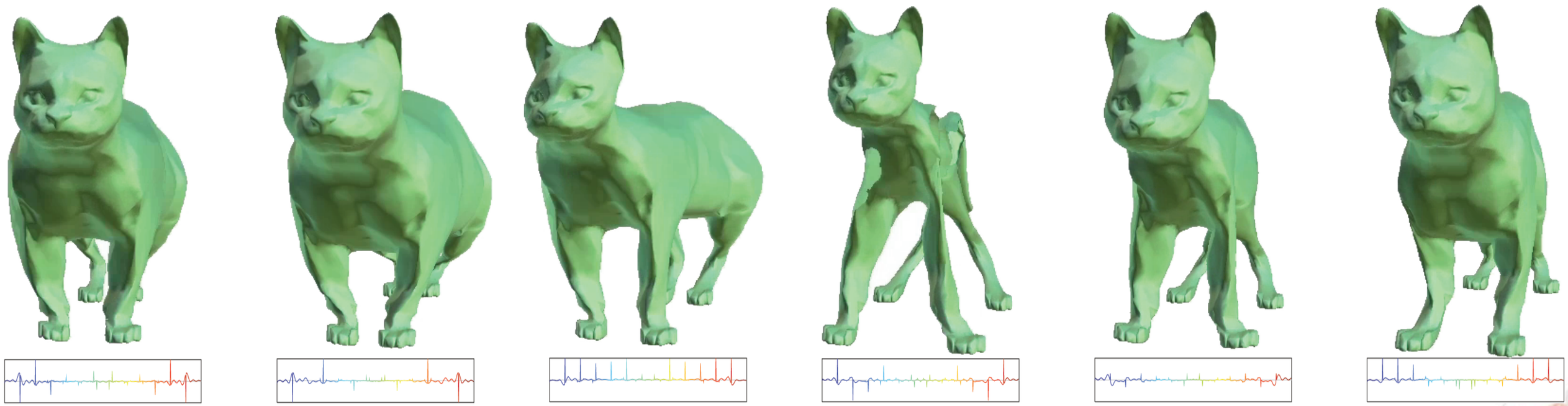
procedural shapes from sound analysis

Mixing Both Approaches



3d animations from music

Contributions



sound analysis → 3d animation

Contributions

gallery of different effects

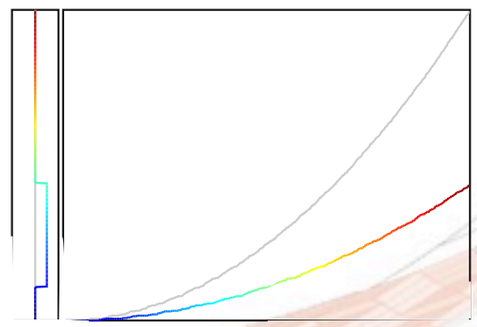
Challenges

Manifold Harmonics Filters in real-time

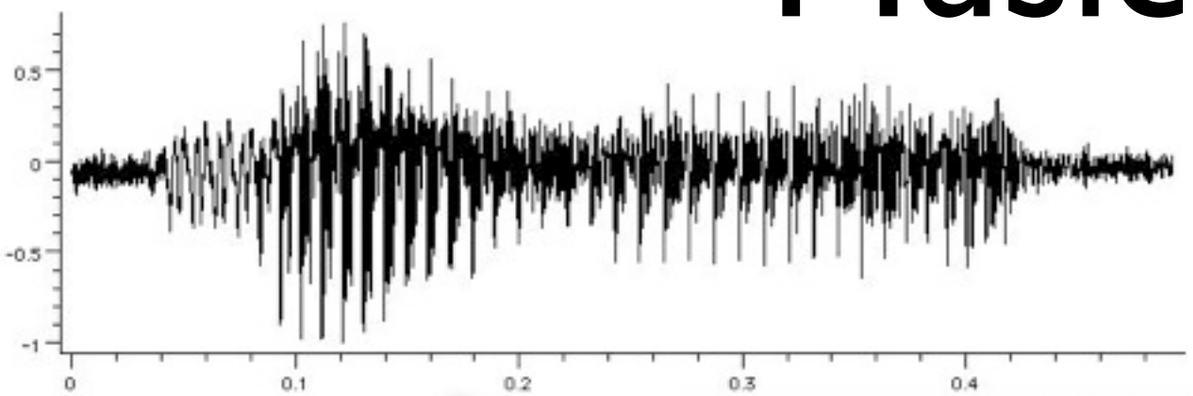
GPU implementation (Gallery x6)

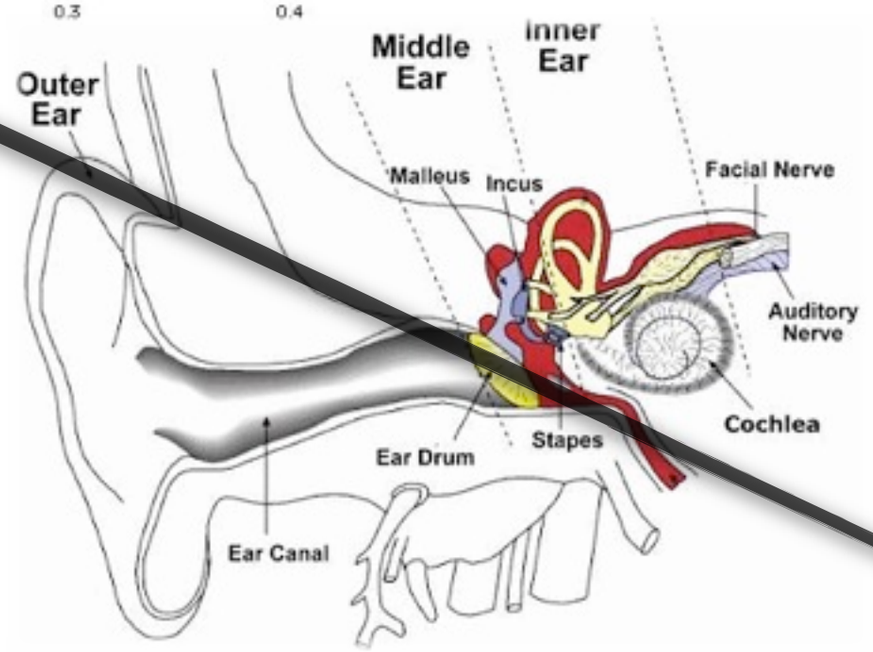
Sound/Mesh transfer function design

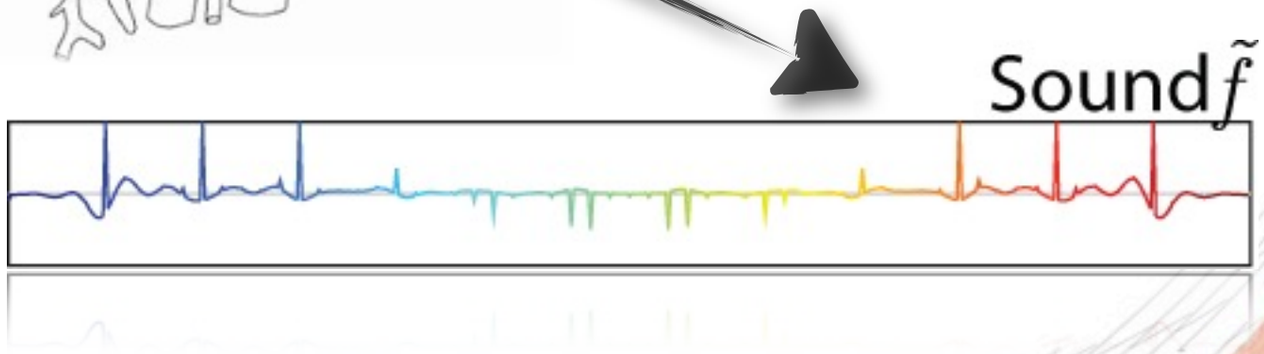
Parameterization for gallery navigation



Music Analysis

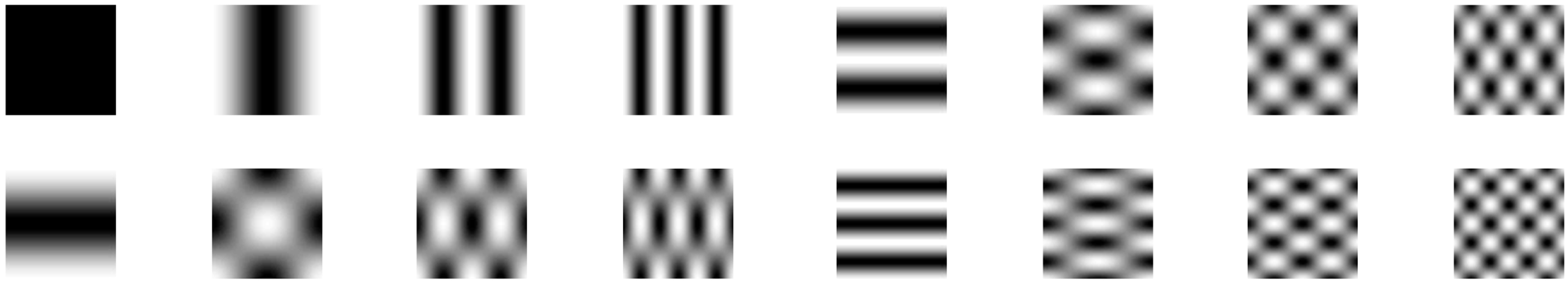


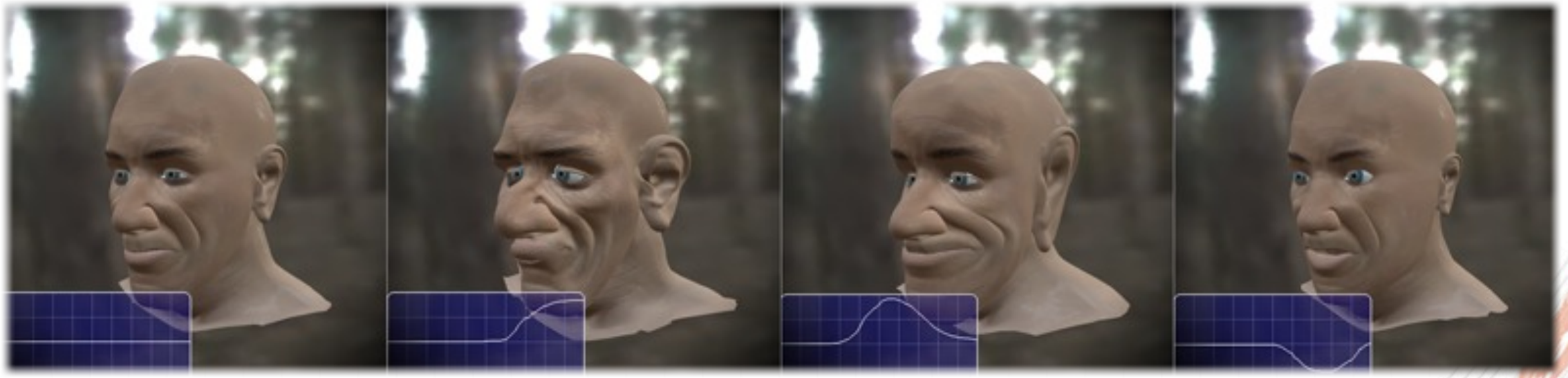




frequency (Fourier transform)

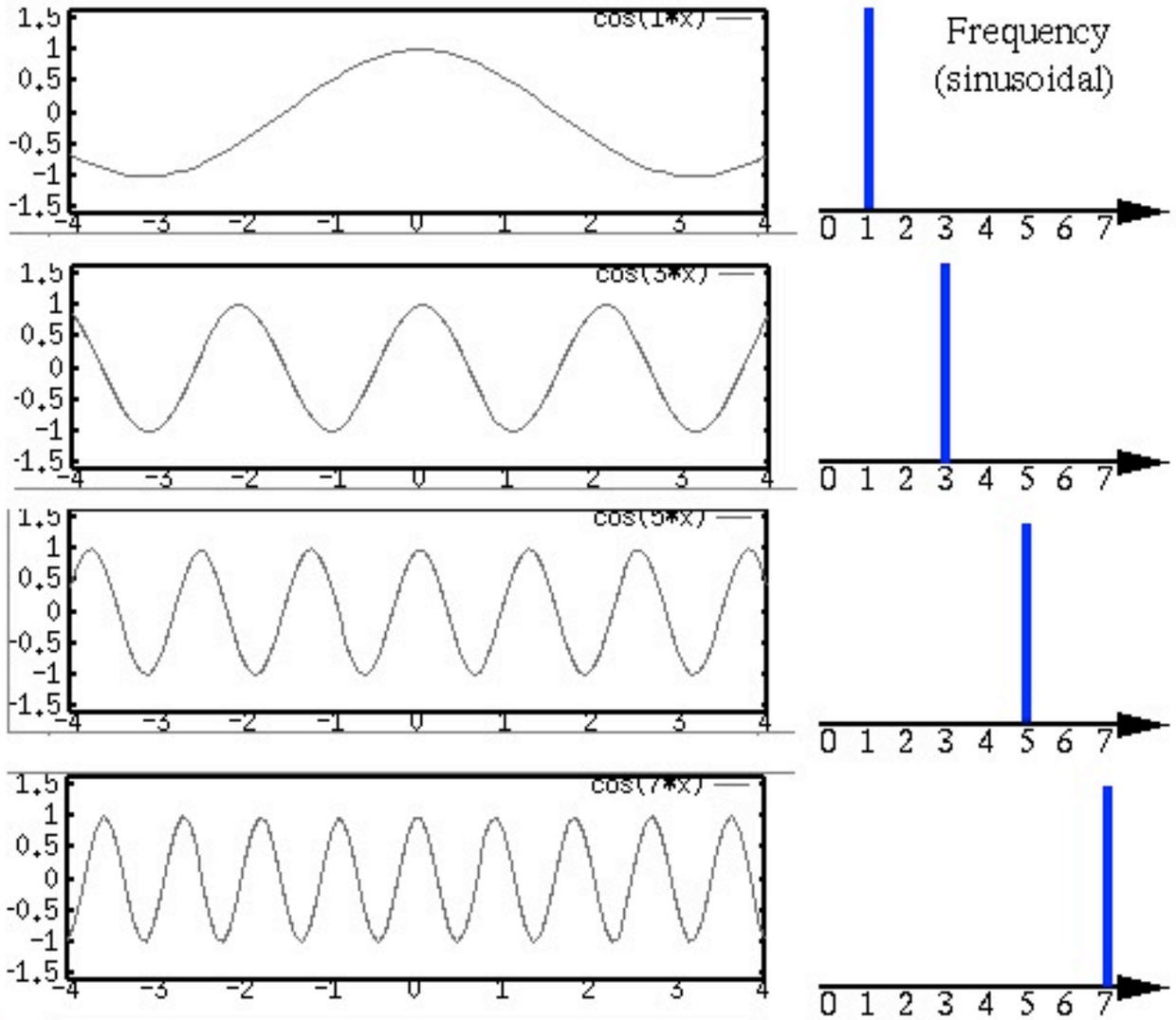
Fourier on meshes?

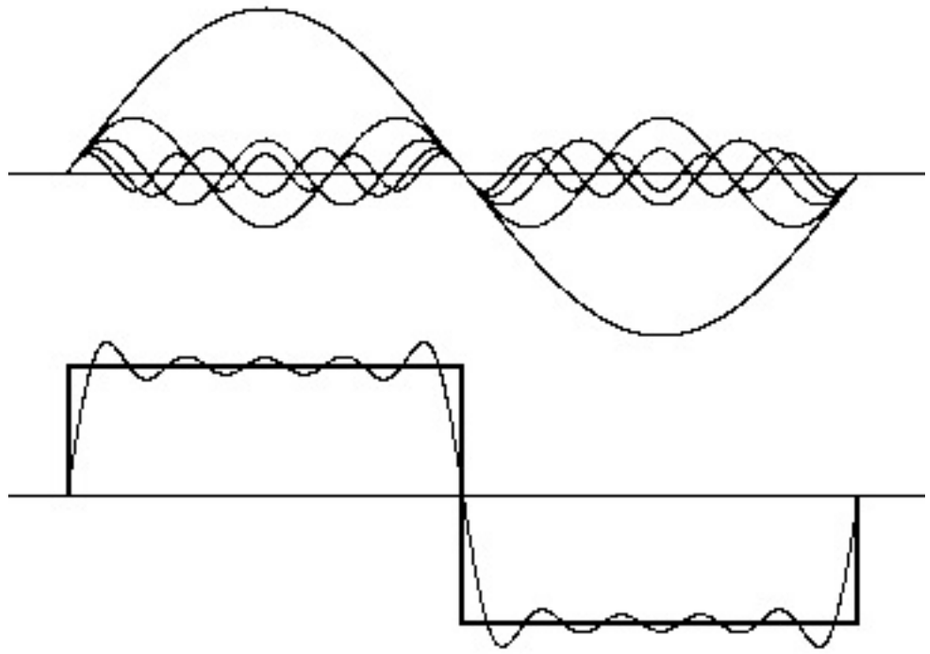




Manifold Harmonics (B. Vallet and B. Lévy, CGF 2008)

Harmonics Analysis





$$f(x) = \sum_{\omega} \hat{f}(\omega) \sin(\omega x)$$

Fourier = decomposition with sin & cos

Manifold Harmonics

Fourier = decomposition with sin & cos

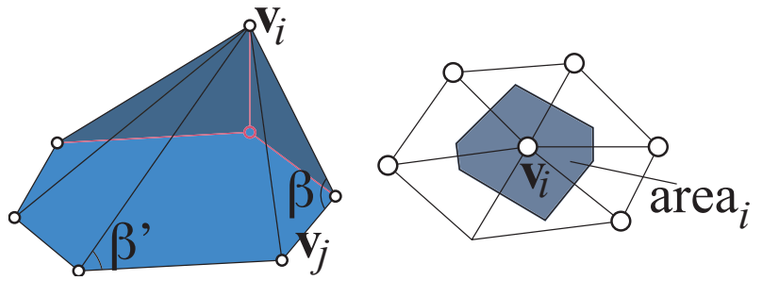
sin & cos: eigenvectors of $-\Delta = -\frac{\partial^2}{\partial x^2}$

$$f(x) = \sum_{\omega} \hat{f}(\omega$$

$x) = \omega^2 \sin(\omega x)$

one variable

Δ on a mesh is a matrix!



$$\Delta_{ij} = -\frac{cot\,(\beta_{ij}) + cot\,(\beta'_{ij})}{\sqrt{area_i \cdot area_j}}$$

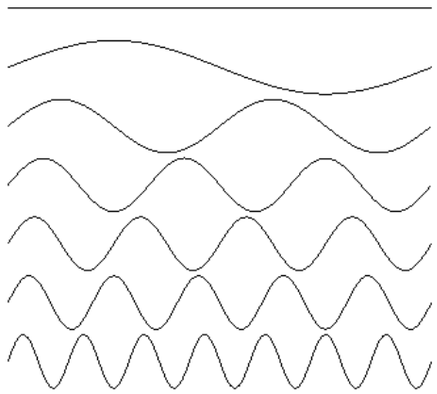
$$\Delta_{ii} = -\sum_{j} \Delta_{ij}$$

$$f(v) = \sum_{\omega} \hat{f}(\omega$$

$) = \lambda \mathbf{H}_\lambda(v)$

mesh

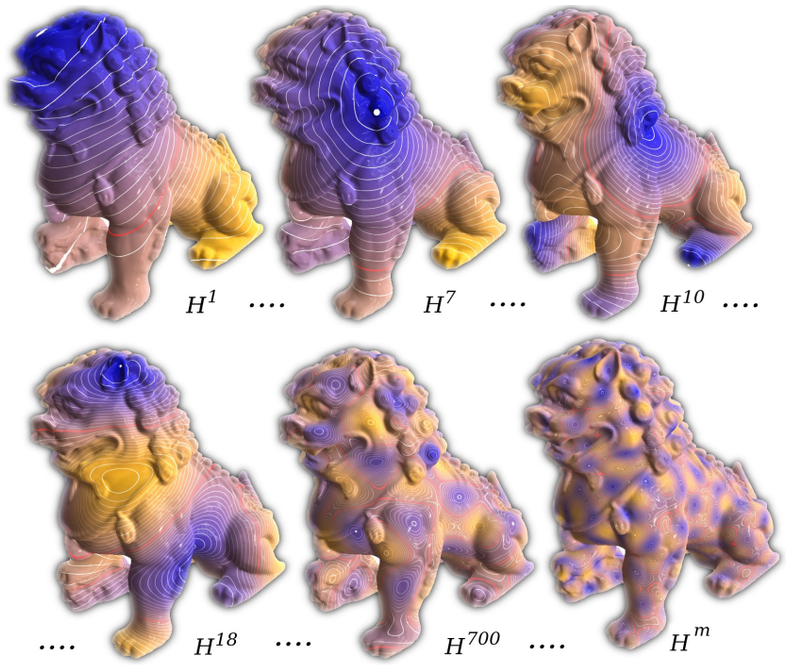
Manifold Harmonics



sin(kx)

on

=



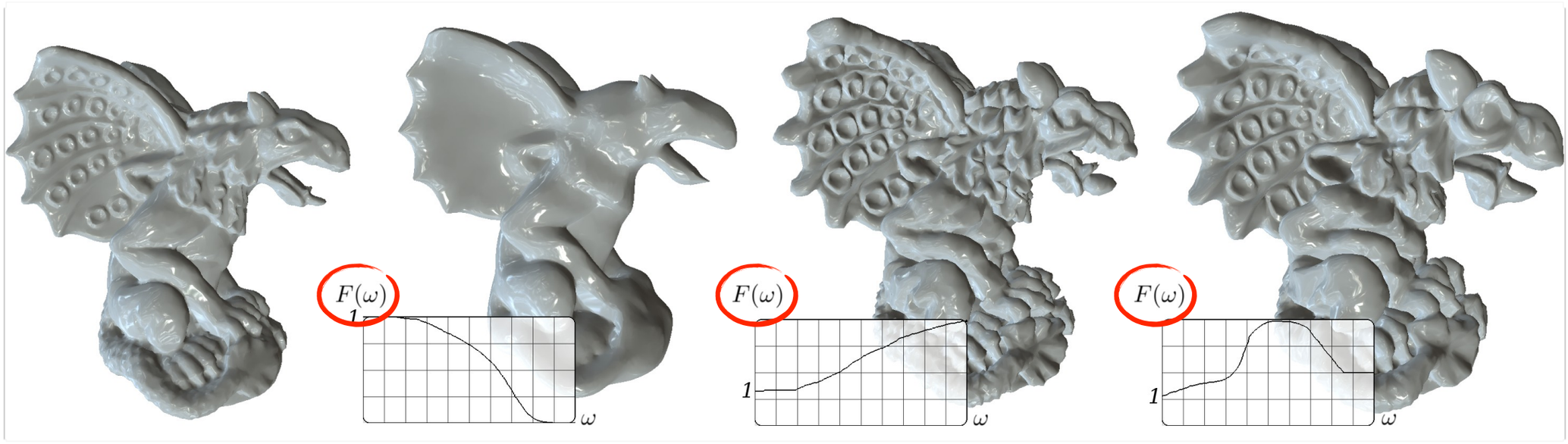
$$f(x) = \sum_{\omega} \hat{f}(\omega) \sin(\omega x) \qquad f(v) = \sum_{\omega} \hat{f}(\omega)\ \mathbf{H}_{\sqrt{\omega}}(v)$$

Harmonics Filters

$$f(x) = \sum_{\omega} \hat{f}(\omega) \sin(\omega x)$$

$$f(v) = \sum_{\omega} \hat{f}(\omega)\ \mathbf{H}_{\sqrt{\omega}}(v)$$

Manifold Harmonics Filters



$$f(v) = v^x$$

$$f(v) = v^y$$

$$f(v) = v^z$$

Vallet, Lévy 2008

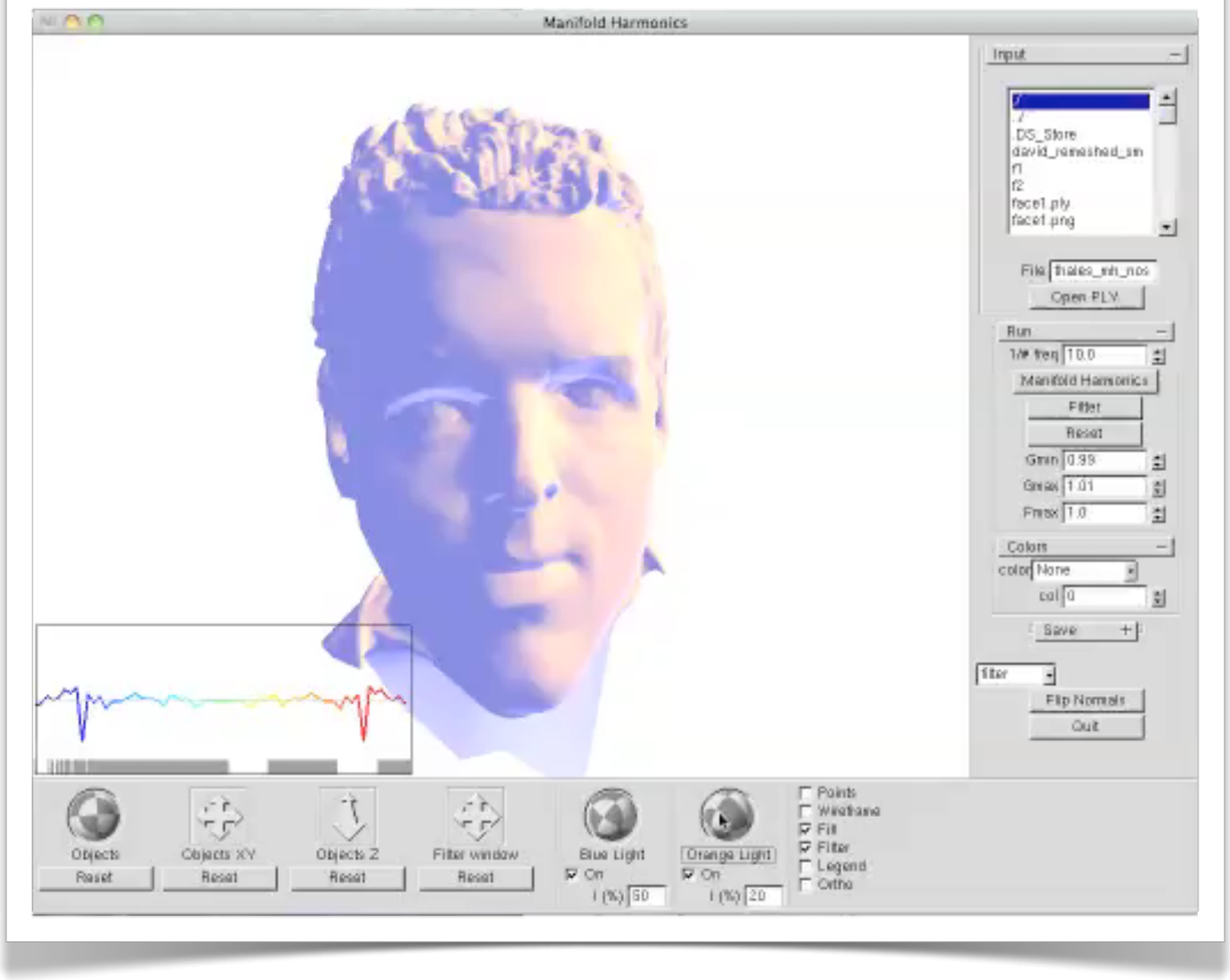
$$f(v) = \sum_{\omega} \hat{f}(\omega)\, \mathbf{H}_{\sqrt{\omega}}(v)$$

$$\hat{f}(\omega)$$

$$\varphi(k) \cdot \begin{cases} \hat{x} \\ \hat{y} \\ \hat{z} \end{cases}$$

$$f(x) = \sum_{\omega} \hat{f}(\omega) \sin(\omega x)$$

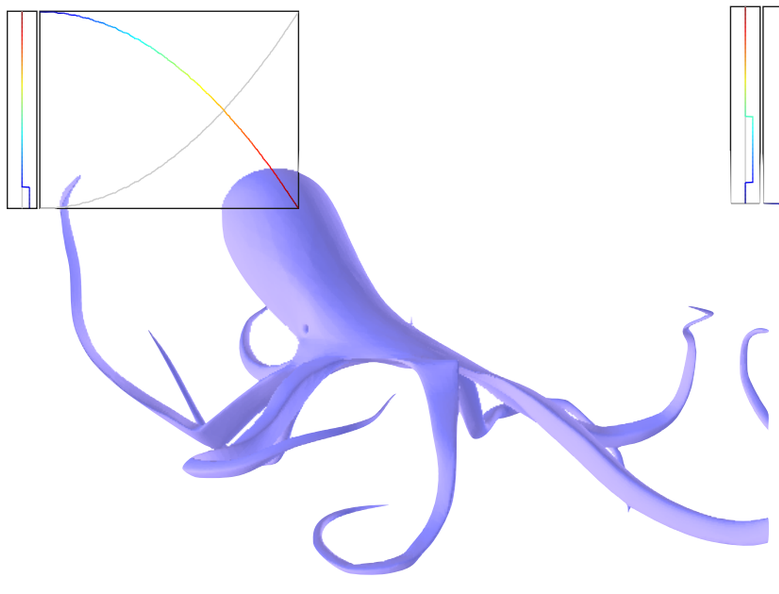
Tuning Filters

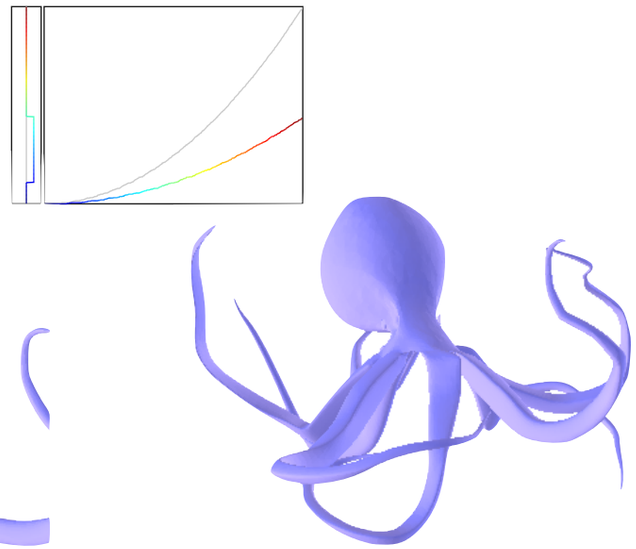


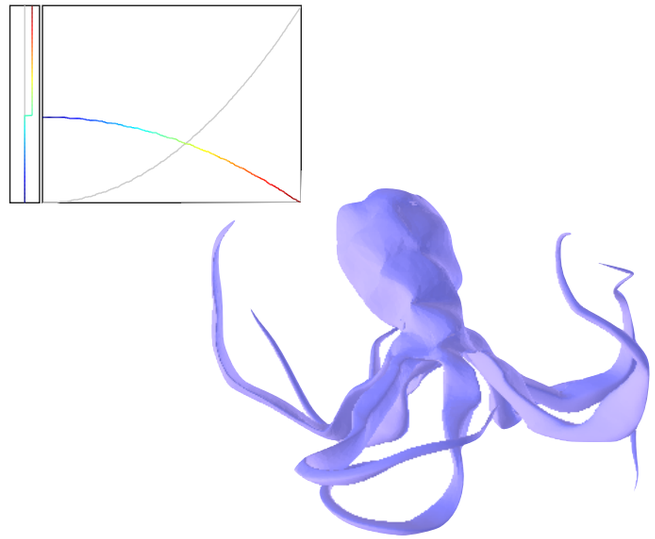
sound amplitudes → filter

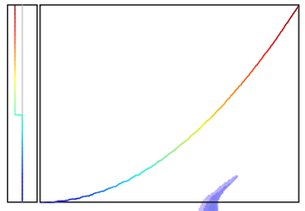
$$1 + \hat{f}(\omega) = \varphi(k)$$

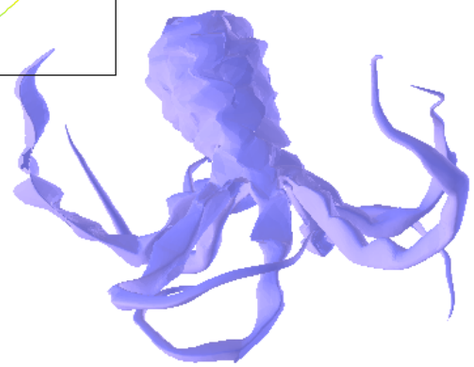
Variations...

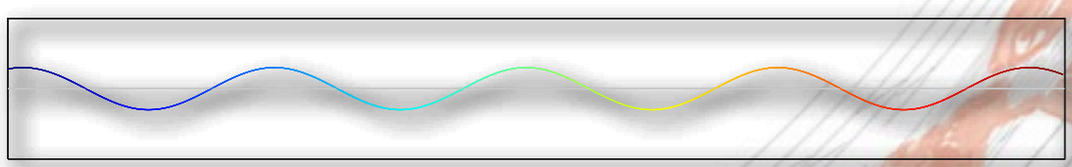


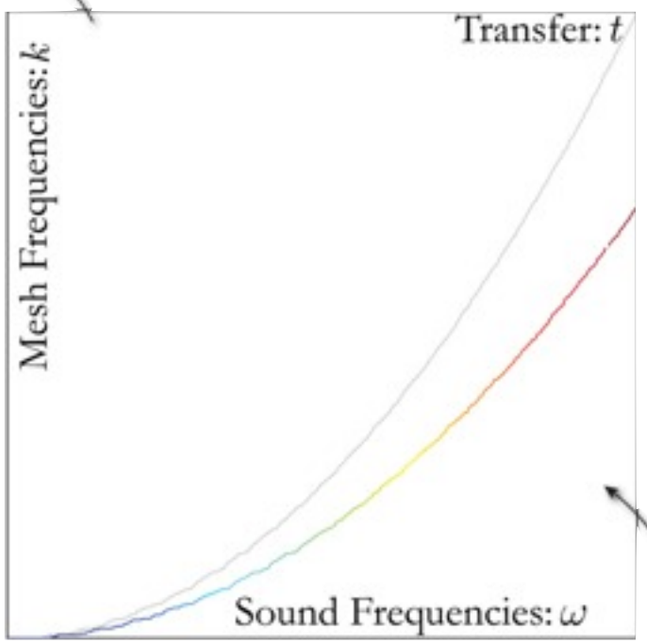








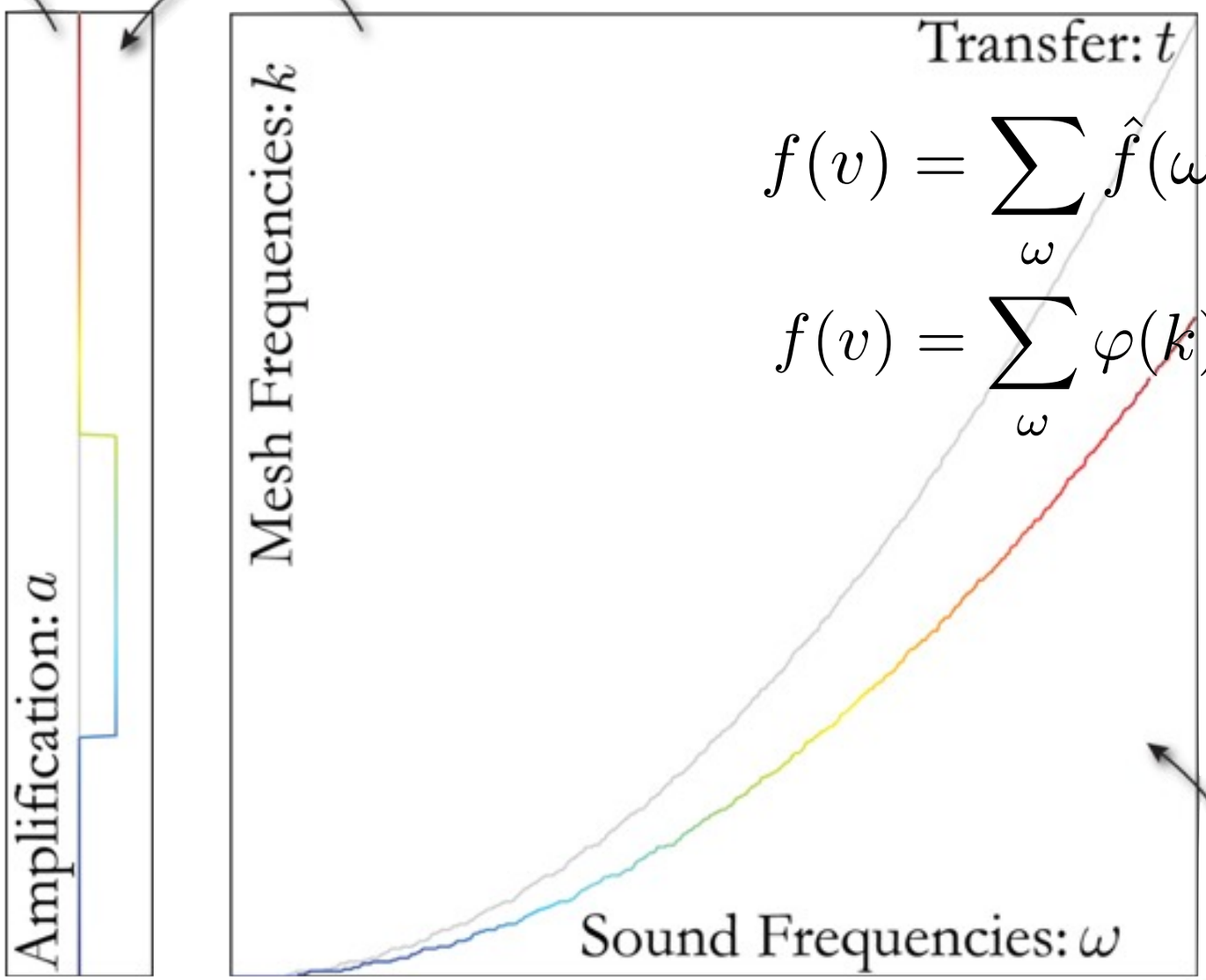




Sound/Mesh Transfer

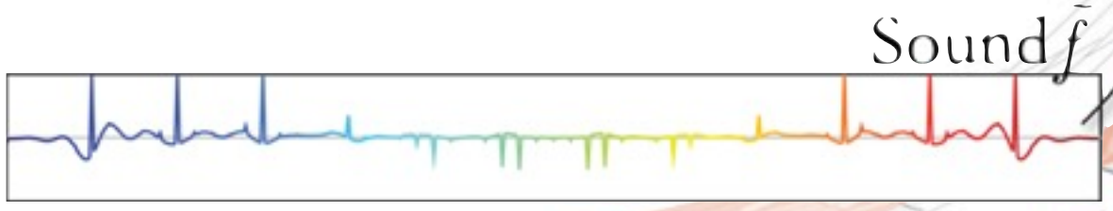
$\varphi(k)$

Deformed mesh



$$f(v) = \sum_{\omega} \hat{f}(\omega)\ \mathbf{H}_{\sqrt{\omega}}(v)$$

$$f(v) = \sum_{\omega} \varphi(k)\hat{x}(v)\ \mathbf{H}_k(v)$$



$\hat{f}(\omega)$

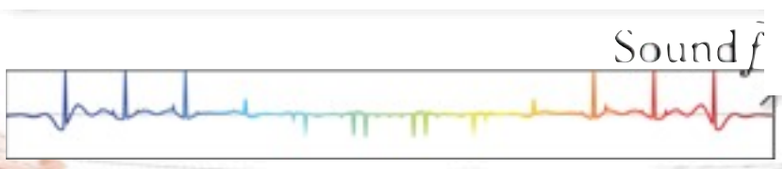
Sound/Mesh Transfer

$\varphi(k)$

$$\varphi(k) = a(k) \cdot \left(\sum_{\omega \in t^{-1}(\{k\})} \hat{f}(\omega) \right) + 1$$

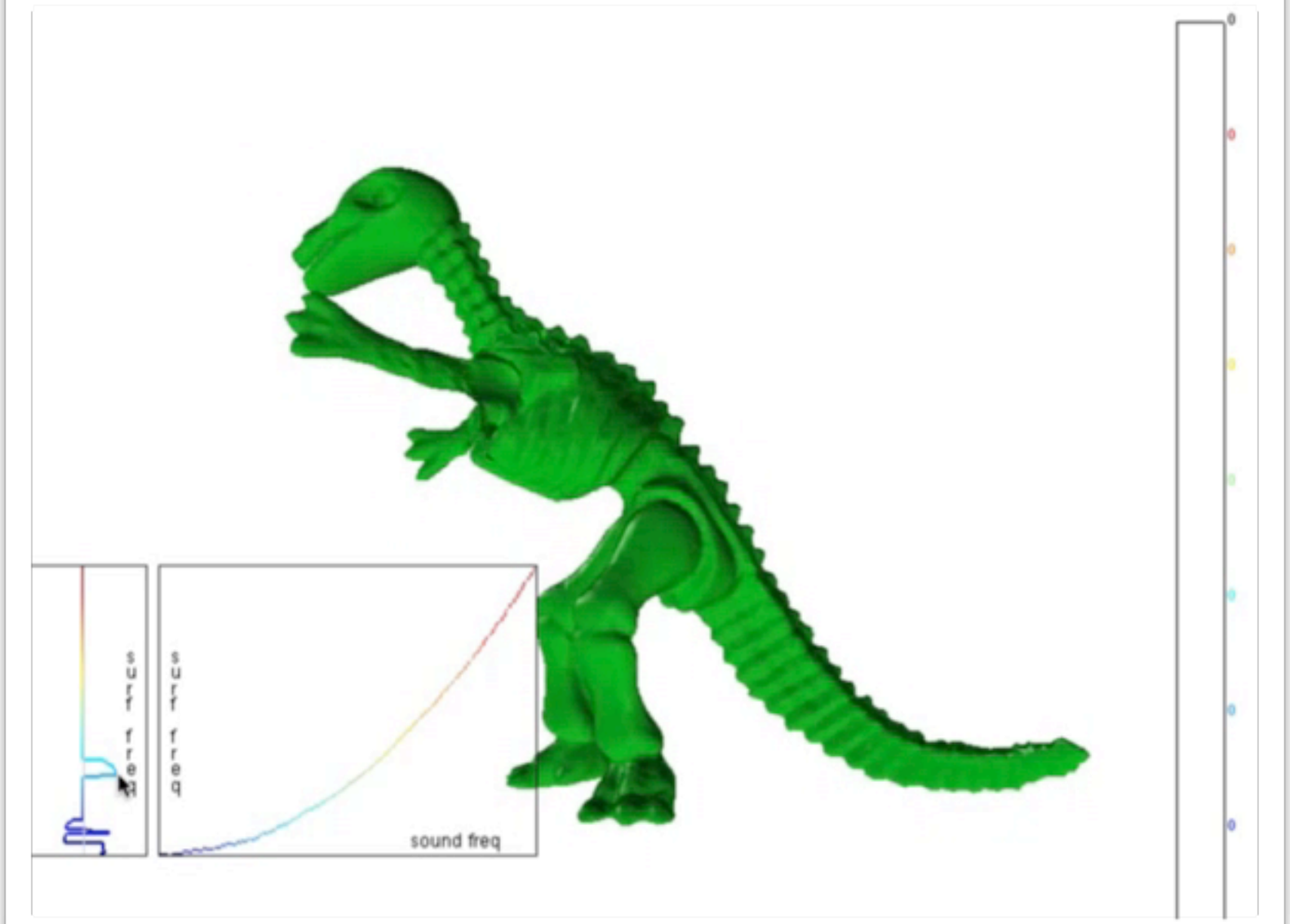
$$\varphi(k) = \sum_{\omega \in t^{-1}(\{k\})} \hat{f}(\omega)$$

k



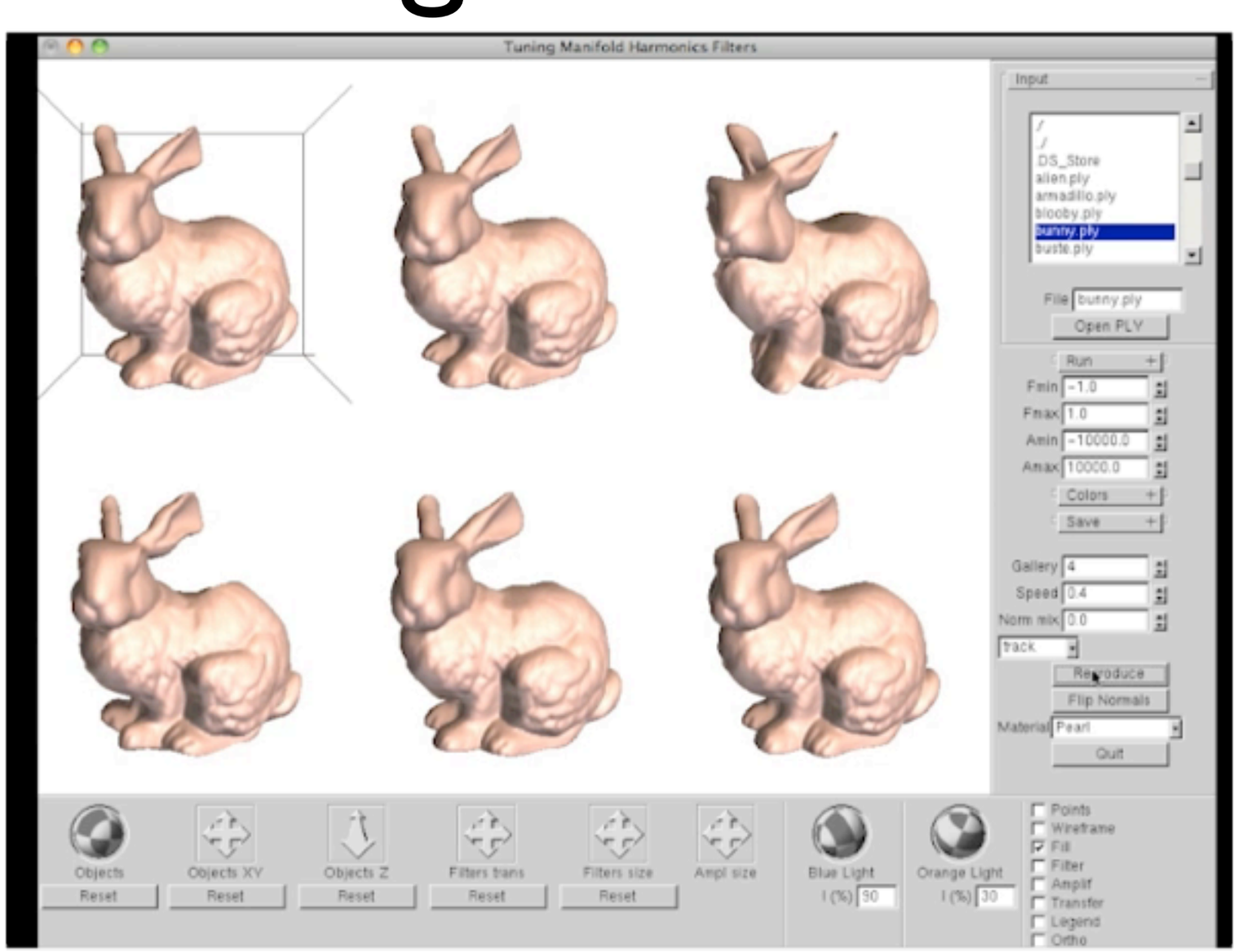
$\hat{f}(\omega)$

Direct Transfer Editing



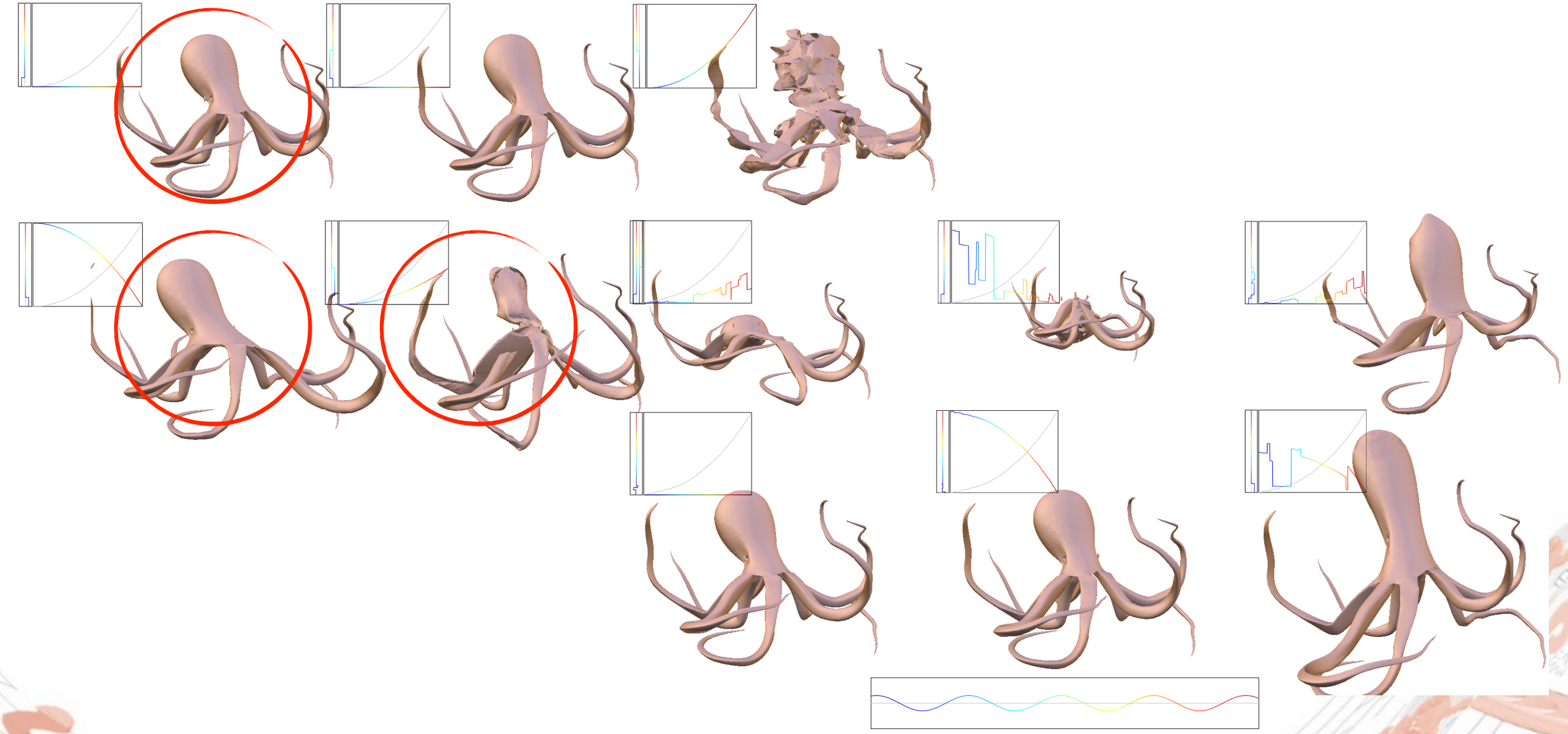
composition of $t(\omega)$ and $a(k)$

Design Galleries



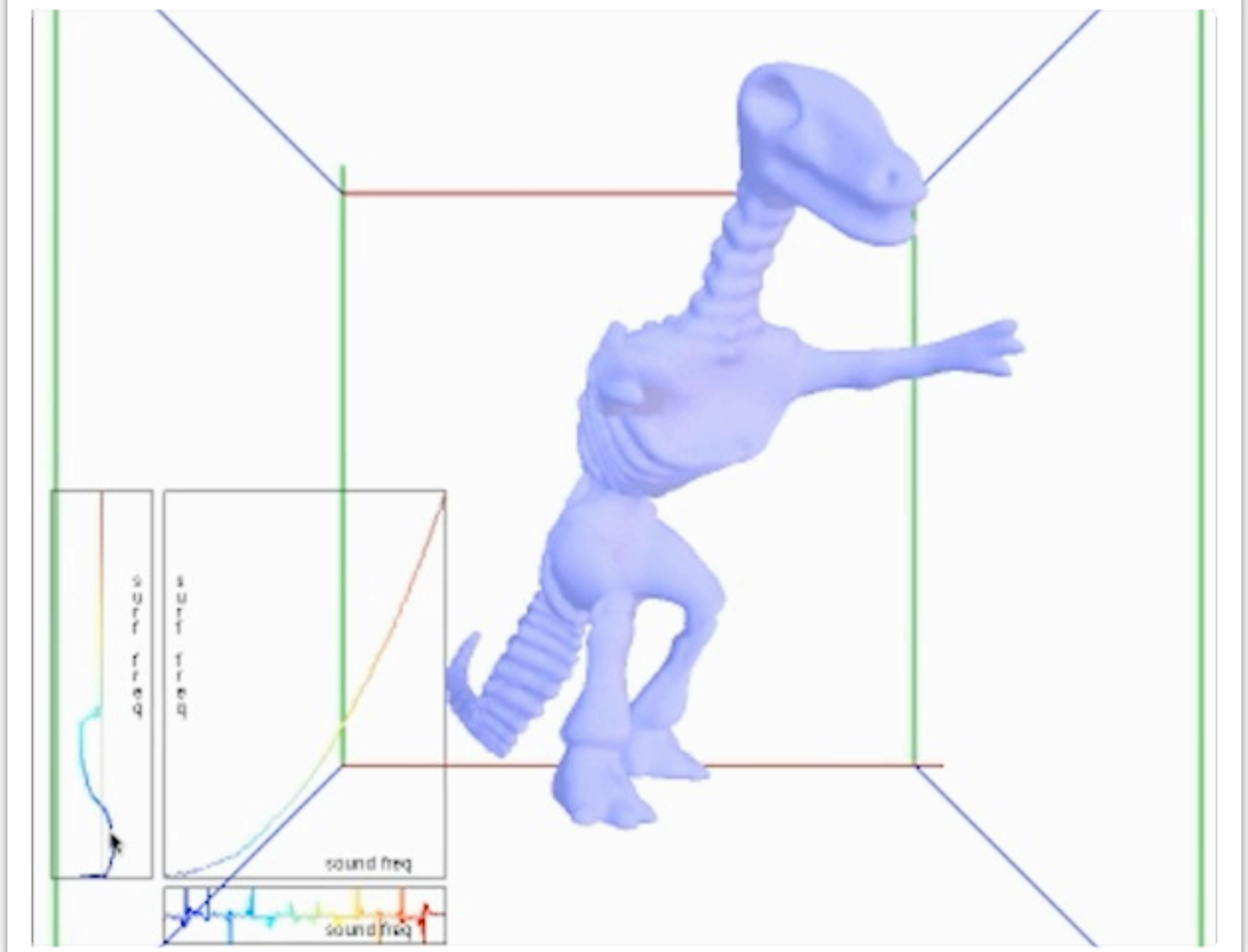
let the user choose!

Gallery Reproduction



independent linear combinations of $t(\omega)$ and $a(k)$

Realtime Galleries



several filters in realtime!

independent *a* and *t* ➙ less items

GPU Implementation

Pre-processing

Manifold Harmonics Basis computation

Laplacian on *CPU* ➙ *texture* H_k

Manifold Harmonics decomposition

x,y,z scalars on *CPU* ➙ *texture x,y,z*

Remainder (high frequencies)

d vectors on *CPU* ➙ *texture d*

Filter

Transfer function

CPU ➙ *texture* φ

Fragment program + render to vb

```
uniform sampler1D x̃ỹz̃;
uniform sampler2D d_xyz;
uniform sampler3D H_k;
uniform sampler1D φ;
uniform float δk;

void main() {
    vec3 texcoord = gl_TexCoord[0].stp ;
    vec3 pos = texture2D(d_xyz,texcoord.st).xyz ;
    for( float k=0.0; k ≤ 1.0; ) {
       texcoord.p = k ;
       vec4 H = texture3D(H_k, texcoord);
       vec4 f = texture1D(φ, k);
       vec3 x̃ỹz̃_0 = texture1D(x̃ỹz̃, k).xyz ; k += δk ;
       vec3 x̃ỹz̃_1 = texture1D(x̃ỹz̃, k).xyz ; k += δk ;
       vec3 x̃ỹz̃_2 = texture1D(x̃ỹz̃, k).xyz ; k += δk ;
       vec3 x̃ỹz̃_3 = texture1D(x̃ỹz̃, k).xyz ; k += δk ;
       pos += f[0] * H[0] * x̃ỹz̃_0 + f[1] * H[1] * x̃ỹz̃_1 +
              f[2] * H[2] * x̃ỹz̃_2 + f[3] * H[3] * x̃ỹz̃_3 ;
    }
    gl_FragColor.rgb = pos.xyz ;
}
```

GLSL (portability)

Textures coordinates: (x,y) ➙ vertex index, z ➙ k index.

Rendering Effects

Normal enhancement

Size-independent transfer function

Beat detection

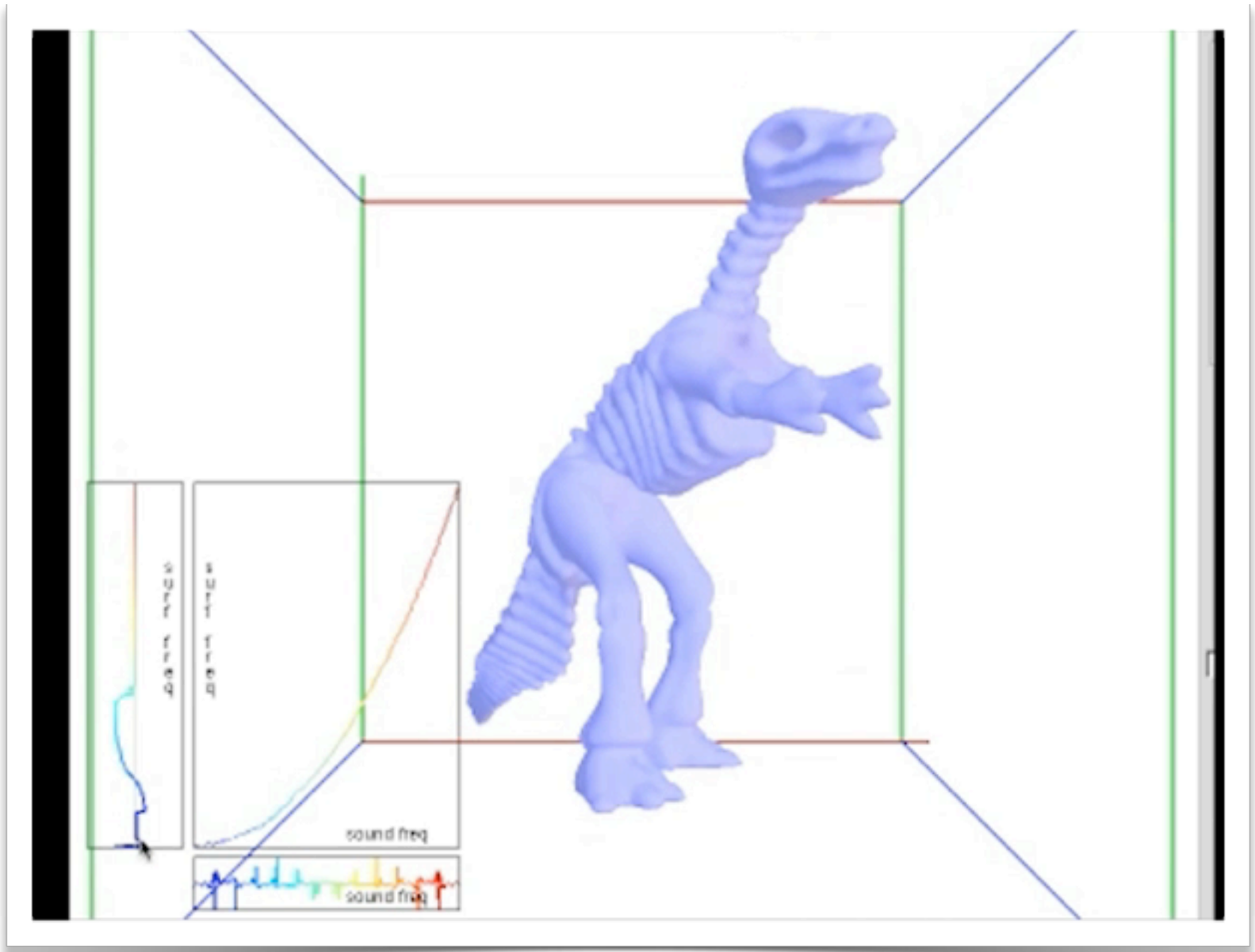
Frequency decay

original N

recomputed N

mixed N

Results



synchronized 3d animation

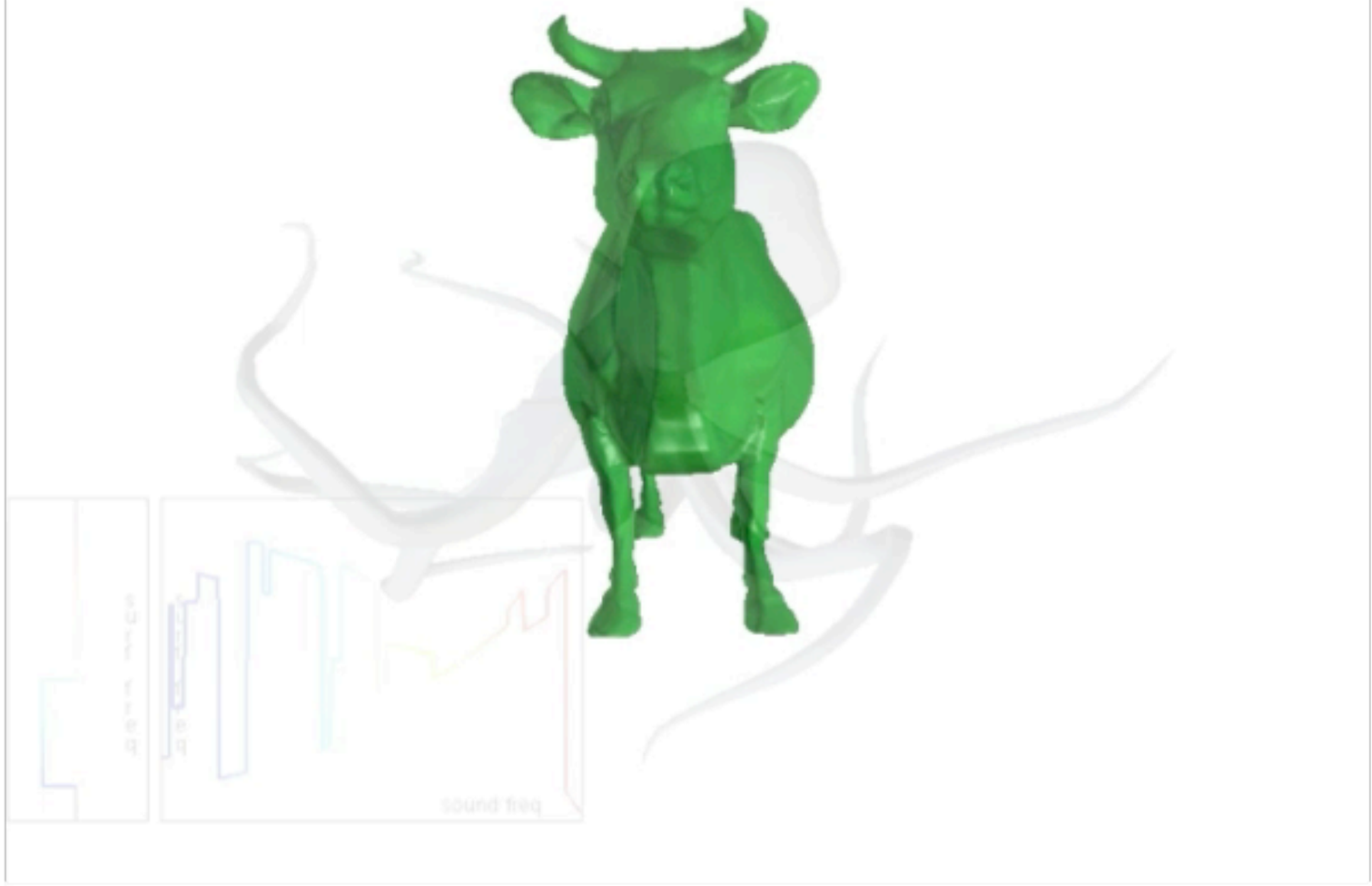
Limitations

Harmonic deformation (preserves texture, but...)

Current GPU's allow 6 to 9 items in a simple gallery

Very raw sound analysis

Next Steps



Use of tessellation shader or Nvidia-specific CUDA?

Better music analysis / camera & lighting control

SIBGRAPI 2011

XXIV Conference on Graphics, Patterns and Images

Maceió - Alagoas - Brazil

August 28th to 31st, 2011

Tutorials
Technical Papers
Technical Posters
Undergraduate Work
Theses and Dissertations
Petroleum
Graphics Processing Education
Video Festival

www.im.ufal.br/evento/sibgrapi2011

Organization: Instituto de Matemática UFAL

Promotion: SBC Sociedade Brasileira de Computação

Cooperation: Eurographics

SIBGRAPI 2011

XXIV Conference on Graphics, Patterns and Images

The XXIV Conference on Graphics, Patterns and Images, Sibgrapi 2011, will be held in the beautiful Maceió, Alagoas, Brazil, between August 28th and 31st. Sibgrapi 2011 is being organized by the Institute of Mathematics of the Universidade Federal de Alagoas (UFAL).

SIBGRAPI started in 1988 and is the most important Brazilian Meeting in Computer Graphics, Image Processing and Computer Vision. It is annually promoted by the Sociedade Brasileira de Computação (SBC). The Sibgrapi proceedings are available on-line at IEEE Xplore since 1997 and have also been published by IEEE Computer Society Press. Since 2009, Sibgrapi further cooperates with Eurographics Association.

We cordially invite you to submit your work to Sibgrapi 2011 and join us in the conference.

General Chairs
Adelailson Peixoto (UFAL)
Dimas Martinez (UFAL)
Thales Vieira (UFAL)

Program Chairs
Thomas Lewiner (PUC-Rio)
Ricardo Torres (Unicamp)

Technical Posters
Aristófanes Corrêa (UFMA)
Marcelo Siqueira (UFRN)

Undergraduate Work
Herman Gomes (UFCG)
Silvio Melo (UFPE)

Theses and Dissertations
George Darmiton (UFPE)
Maria Andréia F. Rodrigues (UNIFOR)

Video Festival
Fátima Sombra (UFC)
Vinicius Mello (UFBA)

Tutorials
Bruno M. Carvalho (UFRN)
Joaquim Bento (UFC)

Petroleum
Eduardo Setton (UFAL)
William W. M. Lira (UFAL)

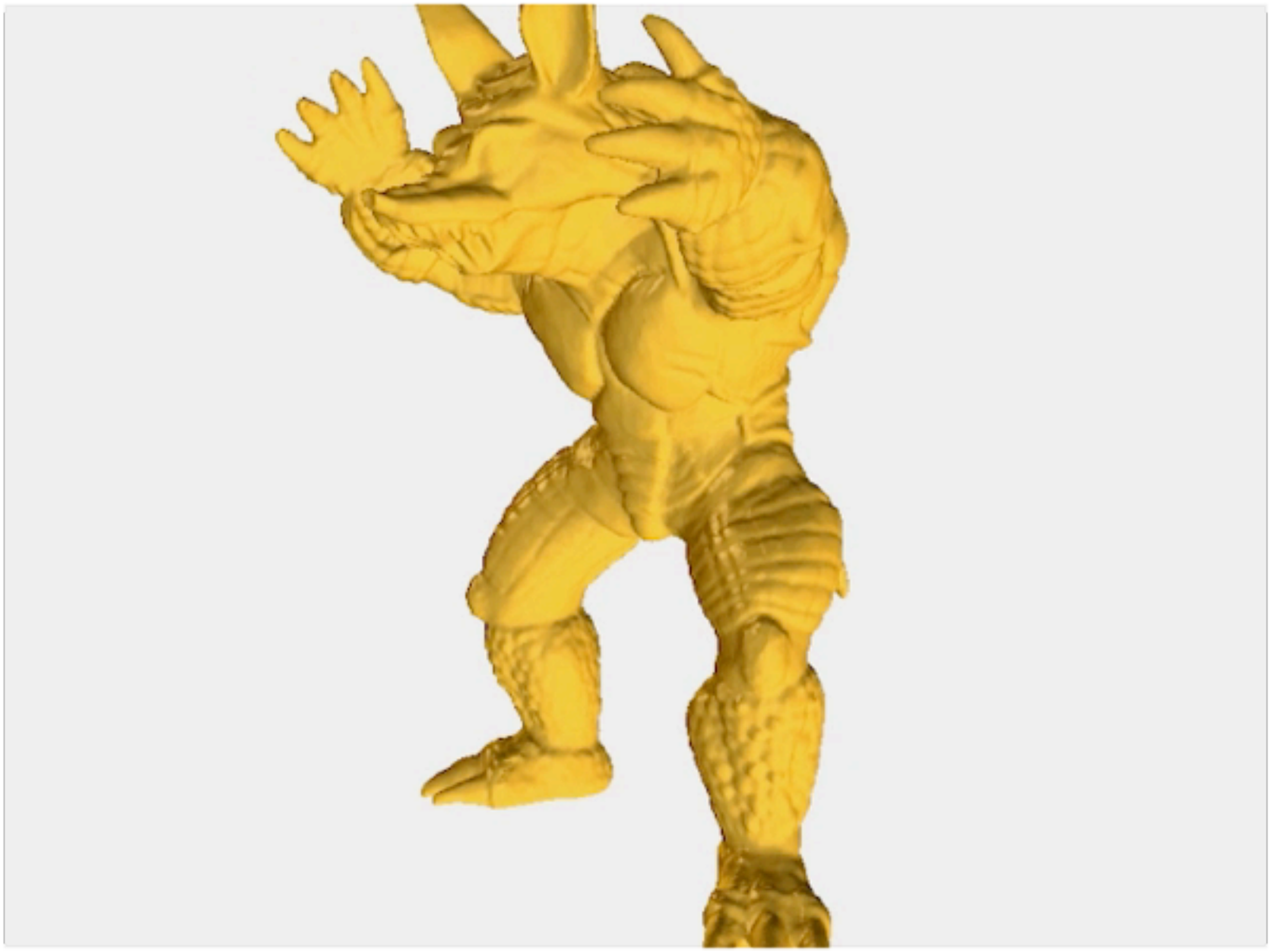
Graphics Processing Education
Veronica Teichrieb (UFPE)

www.im.ufal.br/evento/sibgrapi2011

Thank you for your attention!

http://www.matmidia.org/tomlew
http://www.matmidia.org/research/2010/563/
http://www.youtube.com/watch?v=Gf-vgo2B12A

Thanks to: Brazilian Ministry of Science / CNPq Productivity Scholarship,
Rio de Janeiro's State / FAPERJ Young Scientist of our State Program,
Alagoas State / FAPEAL Summer School Funding
PUC-Rio / Research Grant
The many authors...



Chair: questions?