Interactive 3d caricature from harmonic exaggeration

Thomas Lewiner, Thales Vieira, Dimas Martinez, Adelailson Peixoto, Vinícius Mello and Luiz Velho

MAT. 13/10
Interactive 3d caricature from harmonic exaggeration

Thomas Lewiner, Thales Vieira, Dimas Martinez, Adelailson Peixoto, Vinícius Mello and Luiz Velho


10

December 10th, 2010

http://www.matmidia.mat.puc-rio.br/tomlew/caricature_smi.pdf

Matemática Pura

Matemática Aplicada

Nacional

Internacional

Reconstrução de Objetos Geométricos

Computação Gráfica

Outras Informações Relevantes:

Breve Descrição do Projeto (em caso de projeto novo):

□ Nacional ■ Internacional

□ Matemática Pura ■ Matemática Aplicada
Interactive 3d caricature from harmonic exaggeration

THOMAS LEWINER¹, THALES VIEIRA², DIMAS MARTINEZ², ADELAILSON PEIXOTO², VINÍCIUS MELLO³ AND LUIZ VELHO⁴

¹ Department of Mathematics — Pontificia Universidade Católica — Rio de Janeiro — Brazil
² Institute of Mathematics — Universidade Federal de Alagoas — Maceió — Brazil
³ Institute of Mathematics — Universidade Federal da Bahia — Salvador — Brazil
⁴ Visgraf Laboratory — IMPA — Rio de Janeiro — Brazil

Abstract. A common variant of caricature relies on exaggerating characteristics of a shape that differs from a reference template, usually the distinctive traits of a human portrait. This work introduces a caricature tool that interactively emphasizes the differences between two three dimensional meshes. They are represented in the manifold harmonic basis of the shape to be caricatured, providing intrinsic controls on the deformation and its scales. It further provides a smooth localization scheme for the deformation. This lets the user edit the caricature part by part, combining different settings and models of exaggeration, all expressed in terms of harmonic filter. This formulation also allows for interactivity, rendering the resulting 3d shape in real time.

Keywords: Caricature. Manifold Harmonics. Geometry Processing. GPU. 3d Faces. Shape Modeling.

1 Introduction

Caricature is an illustration technique that exaggerates specific characteristic traits in a portrait of a human subject. Its main goal is to reveal the essence of a person by emphasizing particular aspects that visually identifies the individual. In this way, some features are magnified while other features are attenuated creating a non-realistic personalized impression of the subject [12].

In the western culture, caricature has a long tradition, way back to the Renaissance. Early examples can be found in the works of Leonardo da Vinci. Subsequently, other great artists such as Honoré Daumier specialized in this form of expression. Nowadays, caricatures are present in many types of media, ranging from newspapers and magazines to film and television. It has been traditionally used in political cartoons and then became a powerful means of entertainment in general. Many public figures, such as politicians and movie stars are associated with their caricatured depiction.

Figure 1: Progressive caricature of a scanned head: the user paints the regions to be caricatured, and chooses the amplitude (here $\mu = -6$), scale selection as curve and filter model to use for each part. The differences are obtained by registration with the head of Figure 9 as template.
As a practice, caricature can be considered an art form, not only because the production of an effective caricature requires skill and talent, but mainly because it essentially depends on an interpretation of the reality. In that sense, each caricaturist has a particular style that marks his/her work.

The social importance and appeal of caricature motivates the investigation of this topic in Computer Graphics. In its own way, it configures a remarkably rich area of research because it can be seen both as modeling and as a visualization problem, where perceptual and semantic issues play a fundamental role.

The main challenge in this area is to develop models that are able to sensibly take into account subjective parameters and to implement systems that allow simple intuitive expression. In this perspective, this work introduces an interactive system to create and model three-dimensional caricatures by exaggerating the harmonic differences between a shape and a template.

Historically, the research in computer-based caricature has its origins in the master’s thesis of Susan Brennan in 1982. Since then, the area experiences significant development as reflected by the large number of publications devoted to this subject. Despite of those advances, the basic problems are far from being completely solved and still motivate intense research efforts.

Related Work Computer-based caricature methods can be broadly subdivided into three main categories depending on the principles adopted to model the problem: template-based, extrapolation, and style learning methods. Some of the systems implemented using these methods produce caricatures automatically or semi-automatically, but most of them rely on some form of user interaction.

Template-based methods employ a reference facial shape, which contains specific features that can be emphasized or de-emphasized to different levels in order to produce a caricature. In general, the shape is defined geometrically and warping techniques are used to deform the template geometry interactively. Most of such systems work in two dimensions directly with photographs or with illustrations of a face [1, 2, 11] or parametric three dimensional shapes [10, 19]. The techniques proposed here work directly on scanned 3d shapes.

Extrapolation methods assume that a caricature exaggerates the traits distinguishing the face from the normal one. They resort to an average face which serve as the basis for extrapolating specific features. This kind of techniques works by amplifying the difference of the input face from the mean face. The previously mentioned seminal system of Brennan [5] was based on such principles, also known by illustrators [21]. In recent years, several systems of this type have been proposed [3, 13, 25]. A common way to model the face in such systems is through principal component analysis (PCA) that provides a representation in which the mean face is explicitly defined. These systems also specify exaggeration rules that allow the user to control the caricature effects.

Style-Learning methods, instead of modeling the caricature process by itself, attempt to recreate the mechanisms used by caricature artists. This is typically done using statistical inference techniques that construct a probabilistic model from examples [11, 17, 18, 23]. In this way, it is possible to capture the style of an specific caricaturist using samples of real subjects and the resulting caricatures created by her/him.

Contribution In this work, we propose to use spectral representation of the differences between a template and the 3d shape to be caricatured. We build on the Manifold Harmonics geometry processing [26, 22], since it provides a very intuitive framework for mesh edition, allowing to capture and control meaningful scales of the object through a reduced set of parameters. We derive the extrapolation principle within the harmonic space of the shape to be caricatured, which captures its intrinsic geometry. We propose three different harmonic filters as exaggeration systems. We also use the harmonic basis to provide interactive controls to tune the caricature: a localization control, which specifies which parts of the face should be caricatured; and a scale control to use or exaggerate a subset of the harmonics for the caricature. Our formulation allows for a direct GPU implementation, which gives return on the user’s control in real time. Moreover, a partial result can be used directly in place of the initial shape, letting the user create progressively each part with different methods and parameters (Figure 1).

2 Basics of Manifold Harmonics

In this section, we quickly review the basics of manifold harmonics filtering. The reader will find more details in the original work of Vallet and Lévy [26].

(a) Laplace harmonics

On a discrete mesh $M$ with vertex set $V$, manifold harmonics provide a linear basis $H_k$ for the space of discrete scalar functions $f : V → \mathbb{R}$ defined at the mesh vertices. This basis is built from the eigenvectors of a discrete Laplace operator, $H_k$ being interpreted as the $k^{th}$ fundamental oscillation mode of the mesh. Furthermore, the basis is sorted by increasing frequency, the first eigenvectors corresponding to large scale deformation and the last ones interpreted as fine details or noise.

Figure 2: Geometric elements for the discrete Laplace operator.
In order to construct $H_k$, Vallet and Lévy use the Laplace-De Rham operator derived from Discrete Exterior Calculus, which is given by an $N \times N$ matrix $\Delta$, where $N = \#V$ is the number of vertices of the mesh. Its coefficients $\Delta_{uv}$ are zero if vertices $u$ and $v$ are different and not adjacent, and otherwise:

$$\Delta_{uv} = -\frac{\cot(\beta_{uv}) + \cot(\beta'_{uv})}{\sqrt{a_u \cdot a_v}}, \quad \Delta_{uu} = -\sum_v \Delta_{uv},$$

where $a_v$ is the area of the circumscent dual of vertex $v$, and angles $\beta_{uv}, \beta'_{uv}$ are opposed to edge $uv$ (Figure 2).

(b) Harmonic transform

The above definition of the discrete Laplace operator is symmetric, guaranteeing the existence of real-valued orthogonal eigenvectors $H_k : V \to \mathbb{R}$, satisfying $\Delta H_k(v) = \lambda_k H_k(v)$ and, using Kronecker’s $\delta$ notation, for all vertices $v$ and all frequencies $k$, we have:

$$\sum_{v \in V} H_k(v) \cdot H_l(v) = \delta_{k,l}, \quad \sum_{k=0}^{N-1} a_u \cdot H_k(u) \cdot H_k(v) = \delta_{u,v}.$$

Since $\{H_k, 0 \leq k < N\}$ is a basis, any scalar function $f : V \to \mathbb{R}$ has coordinates on this basis, which we write $f(k)$. With the orthogonality properties, we obtain those coordinates by simple projections:

$$f(v) = \sum_{k=0}^{N-1} a_u \cdot H_k(u) \cdot f(u) \cdot \delta_{u,v}.$$

(c) Scalar Filtering

Using the analogy with Fourier analysis, the square root of the eigenvalue $\sqrt{\lambda_k}$ is interpreted as the frequency of harmonic $H_k$. Given a scalar signal on the mesh $f : V \to \mathbb{R}$, the amplitudes $\tilde{f}(k)$ of each of its frequencies of can be filtered by amplifying each of them by $\varphi(k) \in \mathbb{R}$. The filtered signal $\varphi_f$ is then given by:

$$f_{\varphi}(v) = \sum_{k=0}^{N-1} H_k(v) \cdot \left( \varphi(k) \cdot \tilde{f}(k) \right).$$

As observed in the original work, the high frequencies (i.e. for $k > n$ with $n \ll N$) are expensive to compute, increase the complexity of the filter control, and do not provide significant modeling power at the global scale. We thus use a single factor $\varphi_d$ to amplify all those high frequencies:

$$f_{\varphi}(v) = \sum_{k=0}^{n-1} H_k(v) \cdot \left( \varphi(k) \cdot \tilde{f}(k) \right) + \varphi_d \left( \sum_{k=n}^{N-1} H_k(v) \cdot \tilde{f}(k) \right)$$

$$= \sum_{k=0}^{n-1} H_k(v) \cdot \left( \varphi(k) \cdot \tilde{f}(k) \right) + \varphi_d \cdot d_f(v). \quad (2)$$

The residual $d_f(v)$ of $f$ at each vertex is computed at pre-processing using the complementary expression: $d_f(v) = f(v) - \sum_{k=0}^{n-1} H_k(v) \cdot \tilde{f}(k)$. This way, the last eigenvectors of the basis are never used, and only the restricted basis ($H_k, 0 \leq k < n$) needs to be computed.

3 Interactive 3D Caricature Overview

We propose a caricature system in the extrapolation category, which starts with a reference 3d template (a normal face) and a 3d shape (the face to be caricatured), both represented as triangular meshes (Figure 4). We first compute, for each vertex $v$ of the shape, a corresponding point $x'(v)$ in the template (section 6). We decompose their differences in the frequency domain, using the harmonic basis of the shape, since we want to preserve and exaggerate its features, and not those of the template. We propose three different representations of the differences between the shape and the template: coordinate-wise $x^3$, as a normal displacement $\xi$, or as a distance minimizing filter $\phi$, which are described next.

![Figure 3: Workflow of our interactive 3d caricature system: during preprocessing, the correspondences between the shape and the template and the harmonic decomposition of the shape are computed, and then the harmonic representations of their differences. The user then interacts by selecting morphing, localization and scale controls $\mu, \chi, \varphi$, visualizing the results of those controls in realtime. The user then validates one of the three harmonic exaggeration models, and its resulting caricature is quickly reused in place of the new shape.](image)

The caricature is created by the user through several controls (section 5): a morphing parameter $\mu$, which amplifies the differences; a localization function $\chi(v)$, which specifies the region of the shape to be deformed; and a scale control $\phi^\mu(k)$, which selects and eventually amplifies the frequencies used for the caricature. The localization function $\chi(v)$ serves as a blending between the shape and the deformed region, and is obtained from the characteristic function of that region through filtering, also using the harmonics basis.
Figure 4: Vertex-wise correspondences between a reference template (left) and the original shape to be caricatured (right).

The computation of the caricature for each representation is performed on the GPU, with an exact normal computation (section 2), allowing the display of the three resulting caricatures in real time with high quality. Finally, an optimized reprocessing allows using the result of one caricature in place of the shape, allowing to caricature each part with different parameters, in a layer by layer, progressive manner. The schematic representation of the whole process is illustrated in Figure 3.

4 Harmonic Exaggeration

In this section, we propose three harmonic representations for the differences between the shape and the template. We denote by \(\mathbf{x}(v) = (x(v), y(v), z(v)) \in \mathbb{R}^3\) the position of vertex \(v\) of the shape, and \(\mathbf{x}'(v)\) the corresponding point in the template. The \(k\)-th harmonic of the shape is denoted \(H_k : V \rightarrow \mathbb{R}\).

Given a morphing parameter \(\mu\), and a representation \(\mathbf{X}(k)\) of the differences, we compute the coordinates \(\mathbf{x}_\mu(v)\) of our exaggerated shape using the generic form:

\[
\mathbf{x}_\mu(v) = \mathbf{x}(v) + \mu \cdot \left( \sum_k H_k(v) \cdot \mathbf{X}(k) \right). \tag{3}
\]

This ensures that \(\mathbf{x}_{\mu=0}(v) = \mathbf{x}(v)\), and that \(\mathbf{x}'(v)\) is approximated by \(\mathbf{x}_{\mu=1}(v)\). The caricature is obtained for \(\mu < 0\), exaggerating away from the template (Figure 5).

(a) Coordinate-wise filter

The simplest approach looks for a strict morphing:

\[
\mathbf{x}_{\mu}(v) = \mathbf{x}(v) + \mu \cdot \left( \sum_{k=0}^{n-1} H_k(v) \cdot \mathbf{X}(k) \right) + \mathbf{d}(v) \tag{4}
\]

This ensures that \(\mathbf{x}_{\mu=0} = \mathbf{x}\) and \(\mathbf{x}_{\mu=1} = \mathbf{x}'\). Without further control, this corresponds to linear morphing as obtained by direct correspondences [14].

Observe that there is no single filter \(\varphi\) that achieves mapping the three scalar signals \(x(v), y(v)\) and \(z(v)\) of the shape to the template through Equation (2). This is only possible in general with three different filters \(\varphi_x(k), \varphi_y(k)\) and \(\varphi_z(k)\), which are defined here by:

\[
\begin{align*}
\varphi_x(k) &= 1 + \mu \cdot \frac{x(k)}{x'(k)} \\
\varphi_y(k) &= 1 + \mu \cdot \frac{y(k)}{y'(k)} \\
\varphi_z(k) &= 1 + \mu \cdot \frac{z(k)}{z'(k)}
\end{align*}
\]

Such triple filtering allows mapping any mesh to any mesh with the same connectivity, and does not use much of the intrinsic geometry of the shape (Figure 5).
(b) Normal displacement filter

In order to model the shape to template differences through a single scalar signal, we must give up the exact morphing \( x_{\mu = 1}(v) \approx x'(v) \). We propose to approximate the difference \( x'(v) \) by its projection \( \xi(v) \) along the shape normal \( n(v) \in S^2 \) at vertex \( v \): \( \xi(v) = \langle x'(v) \mid n(v) \rangle \). This normal displacement \( \xi : V \to \mathbb{R} \) is a scalar signal, directly representable in the low and high frequencies of the shape by \( \tilde{\xi}(k) \), \( d_\xi(k) \) through Equation (2). Interpreting \( X(v) \cong \xi(v) \cdot n(v) \), the normal displacement exaggeration is derived from Equation (3):

\[
x^\mu_n(v) = x(v) + \mu \cdot \left( \sum_{k=0}^{n-1} H_k(v) \cdot \tilde{\xi}(k) + d_\xi(k) \right) \cdot n(v) \tag{5}
\]

On perfect conditions (6), \( x'(v) = x(v) + \xi(v) \cdot n(v) \), guaranteeing the morphing. The specificity of those conditions means that this filter restricts the deformation, preserving more of the original geometry of the shape (Figure 5). However this formulation generates auto-intersections in the caricature for large \( |\mu| \).

(c) Distance minimizing filter

Our third filter models the caricature directly as a spectral deformation, looking for a single scalar filter \( \varphi, \varphi_d \) that, equally applied to each of the three coordinate signals \( x'(v), y'(v) \) and \( z'(v) \) would best approximate the template. Using a 2-norm, pair-wise distance for approximation measure, this error is expressed as \( \sum_{v \in V} \| x_\varphi(v) - x'(v) \|^2 \), where the optimization variables \( \varphi, \varphi_d \) appear in \( x_\varphi = \sum_{k=0}^{n-1} H_k(v) \cdot (\varphi(k) \cdot \tilde{x}(k)) + \varphi_d \cdot d_x(v) \). Subtracting 1 to each \( \varphi(k) \) and \( \varphi_d \) is equivalent to replacing \( x' \) by \( x^d \) in the measure (see appendix):

\[
\phi^o, \phi^o_d \triangleq \arg \min_{\varphi, \varphi_d} \frac{1}{2} \sum_{v \in V} \| x_\varphi(v) - x^d(v) \|^2
\]

This minimizing actually simplifies due to the orthogonality of the harmonic basis, and its solution is explicitly given by (see appendix):

\[
\phi^o(k) = \frac{\langle \tilde{x}(k) \mid \tilde{x}^d(k) \rangle}{\| \tilde{x}(k) \|^2}, \quad \phi^o_d = \sum_{v \in V} \frac{\langle d_x(v) \mid d_x^d(v) \rangle}{\sum_{v \in V} ||d_x(v)||^2}.
\]

This optimal filter is then exaggerated by morphing parameter \( \mu \) using Equation (3):

\[
x^\mu_\varphi(v) = x(v) + \mu \cdot \sum_{k=0}^{n-1} H_k(v) \cdot \phi^o(k) \cdot \tilde{x}(k) + \mu \cdot \phi^o_d \cdot d_x(v).
\tag{6}
\]

This filter uses few extrinsic geometry of the shape, and thus further restricts the deformation possibilities, for small \( |\mu| \) (Figure 5). In positive terms, it better preserves the intrinsic geometry of the shape even for large \( |\mu| \).

5 Localization and Scale Controls

The above filters are already controlled by the morphing parameter \( \mu \), with the generic formulation of the caricatured coordinates \( x_\mu = x(v) + \mu \cdot \left( \sum_{k=0}^{n-1} H_k(v) \cdot \tilde{x}(k) \right) \), with \( \tilde{x}(k) \) according to the chosen filter (Equations (4), (5) and (6)). We enhance the user control by letting her/him specify which part of the shape (e.g., nose, chick) should be deformed, and how much each scale (i.e., frequency) should contribute to the deformation. Those controls are injected in the generic form of \( x_\mu \).

(a) Localization control

The user specifies the region to be deformed simply by painting on the shape (Figure 6). The vertices picked by the pointer defines a characteristic function on the mesh: \( \chi_{01} : V \to \mathbb{R} \). This function is typically 1 for the painted vertices and 0 otherwise, but an intensity lower than 1 may be specified before picking. The deformation is then restricted to the painted region by: \( x_\mu(v) = x(v) + \mu \cdot \chi_{01}(v) \cdot \left( \sum_{k=0}^{n-1} H_k(v) \cdot \tilde{x}(k) + d_x(v) \right) \).
If restricting the deformation directly to the painted region, the blending by $\chi_{101}$ would lead to cracks. We decompose $\chi_{101}$ to obtain a smoother localization function $\chi$. Since we already computed the harmonics decomposition on the mesh, we obtain a surface Gaussian smoothing $\chi$ interactively by a low-pass filter on $\chi_{101}$:

$$\chi(v) = \sum_{k=0}^{n_k-1} H_k(v) \cdot \tilde{\chi}(k).$$

The cutoff frequency $n_k$ is specified by the user. The caricature is then smoothly localized by:

$$x_{\mu}(v) = x(v) + \mu \cdot \chi(v) \cdot \left( \sum_k H_k(v) \cdot \tilde{X}(k) + dX(v) \right).$$

The user can caricature independently several regions by result on one region as the new shape, and specify a new region on that shape.

(b) Scale control

Our caricature system offers controls to exaggerate rather fine details or larger scale differences, by restricting or amplifying the difference representation to certain frequencies. This is simply done through an independent filter $\Phi(k), \Phi_d$ (Figure 7):

$$x_{\mu}(v) = x(v) + \mu \cdot \chi(v) \left( \sum_k H_k(v) \cdot \tilde{X}(k) + dX(v) \right).$$

Figure 7: The result of Figure 6 with $n_k = 96$ and the normal displacement filter is used in place of the original shape to continue the caricaturing on the chin (left). The frequencies can be used equally (middle) or selecting and amplifying some of them through $\Phi$, here amplifying intermediate frequencies (right) to mark the chin wrinkle.

Combining with the morphing, localization and scale control, we get the final formulations for each filter:

- normal displacement filter (Equation [5]):
  $$x_{\mu}(v) = x(v) + \mu \cdot \chi(v) \left( \sum_{k=0}^{n-1} H_k(v) \cdot \tilde{X}(k) + \Phi_d \cdot dX(v) \right).$$

- distance minimizing filter (Equation [9]):
  $$x_{\mu}(v) = x(v) + \mu \cdot \chi(v) \left( \sum_{k=0}^{n-1} H_k(v) \cdot \phi(k) \cdot \tilde{X}(k) + \Phi_d \cdot p \cdot dX(v) \right).$$

6 Implementation

The above formulations allow for interactive implementation, offering to the user an immediate return from the chosen setting of each control. This is achieved through a GPU implementation of the filters, leaving to the CPU the low-pass filter for the localization control $\chi$, which is very fast since $n_k$ is small. The picking for the scale control $\Phi$ and the definition of $\chi_{101}$ is also handled in CPU. Finally, to correctly illuminate the caricatured shapes, we propose a GLSL program to compute the vertex normals. This geometry shader is not specific to our filters and would apply to any triangular mesh.

(a) Shape / template correspondences

The correspondences computation is a challenging problem, mainly because geometries may differ considerably between the template and the shape. In addition, our filters presume that meshes coordinates are represented in a common global coordinate system, requiring a registration step.

A first option for the correspondences relies on semi-automatic cross-parameterization [14]. A set of matching vertices is obtained by user interaction, and a common, coarse base mesh is constructed on those vertices. The connectivity of the shape is then mapped onto the template through the parameterizations obtained on each element of the base mesh. Then, we align the shape to the template through a rigid-body transformation, computed by minimizing a point-to-point error metric between a set of correspondences. We use the correspondences obtained by the cross-
Interactive 3d caricature from harmonic exaggeration

Figure 9: Progressive caricature of a face model registered to the face of Figure 4. The last step uses a scaled down model as template.

Algorithm 10: Geometry shader for the vertex normal.

7 Results

We experiment our interactive caricaturing tool, letting the user choose the filters that were more expressive to her/him. We first observe that each of the three filters was used, with a preference on the normal displacement filter for smooth shape (Figure 7), distance minimizing for isolated parts (Figures 8, 11 and 12) and coordinate-wise for the final exaggeration (Figure 9).

We also test different correspondences computation methods: cross-parameterization [14], as illustrated in Figures 8, 11 and 12; registration only and scaled down model, both used in the examples of Figures 1 and 2. In particular, we can observe that even relatively poor global correspondences (Figure 8) leads to expressive caricatures (Figures 11 and 12).

Preprint MAT. 13/10, communicated on December 10th, 2010 to the Department of Mathematics, Pontifícia Universidade Católica — Rio de Janeiro, Brazil.
We set the cut-off frequency to $n_X = 96$. We use a strong morphing parameter for the distance minimizing filters: $\mu = -8.0$, while it was much lower $-0.8 > \mu > -2.4$ for the other filters. The scale control $\Phi$ was set by hand, with amplifications from 0 to 2.

The interface achieves above 8 frames per second when displaying the three filters simultaneously, even with the picking and smoothing of $\chi$ which takes in average 39 milliseconds on the models we use (Table 1). The substitution of the shape by one of the results lasts in average 1.2 seconds for the filter computation on CPU and up to 4 seconds for the reprocessing.

**Discussion**

The use of manifold harmonics to model the exaggeration as a filter restricts the deformation to operate on non-localized basis. The reduction of the filters to not too high frequencies further limits the ability to capture details such as wrinkles, while a more careful treatment of high frequencies may [24].

The whole process is also very sensitive to the correspondences accuracy. In particular, using only continuous cross-parameterization reveals the $C^1$-discontinuity of the correspondences for large $|\mu|$ (Figure 8).

<table>
<thead>
<tr>
<th>model</th>
<th>Fig.</th>
<th>verts #V</th>
<th>$H_k$</th>
<th>$\tilde{X}$, $d_X$</th>
<th>user secs</th>
<th>user secs</th>
<th>user fps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Face 1</td>
<td>1</td>
<td>48 783</td>
<td>429</td>
<td>4.4</td>
<td>8.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simple face</td>
<td>4, 7</td>
<td>50 014</td>
<td>422</td>
<td>3.8</td>
<td>8.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Face 2</td>
<td>9</td>
<td>32 525</td>
<td>217</td>
<td>2.5</td>
<td>8.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Max Planck</td>
<td>12</td>
<td>31 293</td>
<td>182</td>
<td>2.3</td>
<td>10.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Egea</td>
<td>11</td>
<td>31 293</td>
<td>183</td>
<td>2.4</td>
<td>9.6</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Performance tests on a 2.8GHz processor with a GeForce GT 9600M with 512MB of RAM. The interaction speed includes the model display, picking, the three filters and the vertex normal update.

**8 Conclusion**

In this work, we propose an interactive tool for 3d caricature modeling. It derives the extrapolation principle [5] in terms of harmonic filtering, allowing for interactivity and providing morphing, localization and scale controls. This work may be improved in several directions, among which enhancing the interface with automatic segmentation to select significant parts to be exaggerated, and including texture in the process to achieve more realistic results. Using non-photo-realistic rendering may also enhance the interface.

**Acknowledgments**

We would like to thank the Max-Planck-Institut für Informatik, Cyberware for the 3d models and the persons who let us use their scanned faces. This work is partially financed by CNPq, FAPERJ and FAPEAL.

**References**


Interactive 3d caricature from harmonic exaggeration

Figure 12: Progressive caricature of the Max Planck model with template from Egea (Figure 8).


Preprint MAT. 13/10, communicated on December 10th, 2010 to the Department of Mathematics, Pontifícia Universidade Católica — Rio de Janeiro, Brazil.
A Distance optimization details

The distance minimizing filter \( \phi^o(k), \phi^o_d \) is defined as the global minimum of the quadratic functional \( F(\varphi, \varphi_d) = \frac{1}{2} \sum_{v \in V} \| \mathbf{x}_\varphi(v) - \mathbf{x}^\delta(v) \|^2 \). First, substituting \( \varphi \) per \( \varphi - 1 \) leads to a simpler form:

\[
\mathbf{x}_\varphi(v) - \mathbf{x}^\delta(v) = \mathbf{x}_{\varphi - 1}(v) - \mathbf{x}^\delta(v)
\]

Using this substitution and the orthogonality properties of the harmonic basis allows to solve this optimization problem explicitly. Without grouping the high frequencies yet, i.e. considering \( \varphi(k) = \varphi_d \) for \( k \geq n \), we can write \( F \) in the harmonic basis.

\[
F(\varphi) = \frac{1}{2} \sum_{v \in V} \| \mathbf{x}_\varphi(v) - \mathbf{x}^\delta(v) \|^2
\]

\[
= \frac{1}{2} \sum_{v \in V} \sum_{k=0}^{N-1} \left( \phi(k) \cdot \mathbf{x}_\varphi(k) - \phi(k) \cdot \mathbf{x}^\delta(k) \right)^2
\]

\[
= \frac{1}{2} \sum_{v \in V} \sum_{k=0}^{N-1} \left( \sum_{l=0}^{N-1} \phi(k) \cdot \mathbf{x}_\varphi(k) - \phi(k) \cdot \mathbf{x}^\delta(k) \right)^2
\]

Grouping the terms by \( v \) and using Equation [1]:

\[
F(\varphi) = \frac{1}{2} \sum_{k,l=0}^{N-1} \left( \sum_{v \in V} \phi(k) \cdot \mathbf{x}_\varphi(k) \cdot \mathbf{x}_\varphi(l) \right)
\]

\[
= \frac{1}{2} \sum_{k,l=0}^{N-1} \left( \phi(k) \cdot \mathbf{x}_\varphi(k) \cdot \mathbf{x}^\delta(k) \right)^2
\]

The high frequency part goes the reverse way of the previous derivation of \( F \):

\[
\frac{\partial F}{\partial \varphi_d} = 0 = \sum_{k,n} \left( \mathbf{x}_\varphi(k) \cdot \mathbf{x}_\varphi(l) - \mathbf{x}^\delta(k) \right)
\]

\[
= \sum_{k,l=0}^{N-1} \left( \sum_{v \in V} \phi(k) \cdot \mathbf{x}_\varphi(k) \cdot \mathbf{x}_\varphi(l) \right)
\]

\[
= \sum_{k,l=0}^{N-1} \sum_{v \in V} \phi(k) \cdot \mathbf{x}_\varphi(k) \cdot \mathbf{x}_\varphi(l) \cdot \mathbf{x}^\delta(k)
\]

The minimum argument \( \phi^o, \phi^o_d \) of \( F \) is then explicitly given by:

\[
\phi^o(k) = \frac{\mathbf{x}_\varphi(k) \cdot \mathbf{x}^\delta(k)}{\| \mathbf{x}(k) \|^2}, \quad \phi^o_d = \frac{\sum_{v \in V} \mathbf{d}_\varphi(v) \cdot \mathbf{d}_\varphi(v)}{\sum_{v \in V} \| \mathbf{d}_\varphi(v) \|^2}.
\]