

Expressive Reproduced Characters for Games

Roberto Cesar Cavalcante Vieira, Creto Augusto Vidal, Joaquim Bento Cavalcante-Neto
Federal University of Ceará

Abstract

Virtual tridimensional creatures are active actors in many types of games. Some of these games require, in addition to quantity, the simulation of kinship and evolution, not only of human character models but also of different types of animals, toon models or other creatures. Some applications also require interactions between isolated populations with well-defined ethnic characteristics. The identification of similar traits between individuals of the same family is crucial to providing increased realism. The main difficulty in these situations is to generate models automatically, in real time, which are physically similar to a given population or family. Another desirable feature is the automatic generation of facial expressions as the character interacts with the environment. In those cases, the difficulty lies in finding a simple mesh adaptation system for different creatures with big differences in shape. In this work, the reproduction of diploid beings is mimicked to produce character models that inherit genetic characteristics from their ancestors, with the possibility to map all genes identifying the origin of each gene, and apply custom facial expressions. With this solution it is possible to create interactive evolution and life-simulation games, genetic educational applications, and many other possibilities.

Keywords: Virtual Characters, Reproduction Simulation, Facial Expression

Author's Contact:

robertoccv@gmail.com
{cvidal, joaquimb}@lia.ufc.br

1 Introduction

In many computer games, it is necessary to populate the environments with virtual characters. Building realistic models that meet the current needs of new technologies and applications has become an ever-present challenge to researchers. Additional challenges are posed by applications, such as life-simulation computer games and internet-based virtual worlds, which require simulation of kinship and interaction between isolated populations of different types of creatures with well-defined ethnic characteristics. Interaction factors, such as generation of expressions as the characters interact in the environment, are also desirable.

In this paper we propose a method for facial expressions application to reproduced virtual characters. At the beginning of the reproduction simulation, the average model of the initial parents is calculated (Figure 1a). With the average model, we define the proportion rules of the face to apply expressions (Figure 1b). With the average model and the proportion rules, we simulate the automatic transfer of traits of the whole body of different types of creatures, through simulated diploid reproduction, with the possibility of mapping the origin of each gene in subsequent generations of characters, with the use of identification parameters (Figure 1c). This is an extension of our previous work [Vieira et al. 2012], where we define an architecture for virtual character reproduction. During the application run, as the new characters are produced, it is possible to apply the rules of proportions and generate automatic facial expressions (Figure 1d). In Section 2, we present some of the most relevant related works. In Section 3, we introduce the adaptation systems used in all methods. In Section 4, we summarize the virtual character reproduction architecture and present some case studies of gene mapping. In Section 5, the architecture of expression generation is introduced and shown in case studies. In Section 6, we present our conclusions.

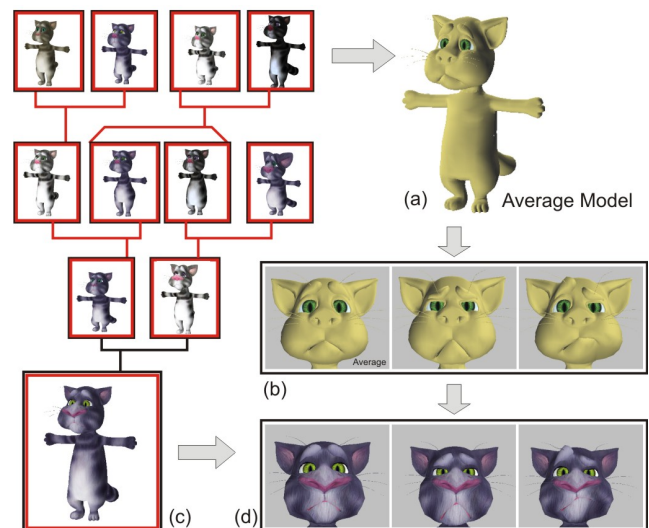


Figure 1: Application of expressions to reproduced characters. (a) Calculation of the average model from the initial parents. (b) Definition of the proportion rules to define expressions, using the average model. (c) Reproduction simulation. (d) Application of expressions to the new models.

2 Related Work

DeCarlo [DeCarlo et al. 1998] devised a system that is capable of generating facial models for virtual reality applications automatically. DeCarlo used B-Spline surfaces to construct a base model, which respected established anthropometric measurements and perturbed those measurements randomly within predefined ranges, to generate other models. Blanz and Vetter [Blanz and Vetter 1999] treated faces as members of a vector space. Thus, starting from a database of faces that were captured through the scanning of real subjects, they used Principal Component Analysis (PCA) to determine a set of face basis vectors that could be linearly combined to generate new faces within that vector space. They could generate realistic models with a large amount of variability. Praun et al. [Praun et al. 2001] used parameterization techniques to combine features of two or more models with different topologies, generating interpolated models. Although their morphing technique showed interesting results, defining the parameters required considerable human effort. Allen et al. [Allen et al. 2003] extended Blanz and Vetter's work to generate the whole human body, combining models with different heights and weights. Seo and Thalmann [Seo and Thalmann 2003] used Bezier patches to develop a base model that could be reshaped by the manipulation of a set of parameters, whose values could be determined by combining specific measurements of scanned 3D models that were stored in a database. Vieira et al. [Vieira et al. 2008], proposed a biologically inspired technique for achieving diversity in the generation of virtual characters for virtual reality applications and games. In the implementation of that technique, different facial models were generated through simulated reproduction of two parental models. The facial model of the descendant was reconstructed from the information that is stored in his genome (a set of genetic characteristics), which contains a set of measurements that control the facial Bezier patches.

Kahler and his coauthors [Kahler et al. 2002] built a base model consisting of anatomical layers (skin, muscles and bones) associated with anthropometric landmarks. With this system of layers, it is possible to deform muscles, generating different expressions, but for each different model, it is necessary to adjust or rebuild the layers system. Kalra [Kalra et al. 1992] used freeform deformation to generate expressions. The mesh of the face is included in a box

containing flexible control points. With the deformation of the box, the mesh of the face is also deformed following the same parameters. Noh and his coauthors [Noh et al. 2000] used the technique of RBF (Radial Basis Functions) to generate facial expressions in real time, to be used in videoconferencing systems. They developed a system for facial deformation based on moving points in an influence region. In their Expression Cloning work, Noh and Neumann [Noh and Neumann 2001] applied similar deformation techniques to transfer different types of expressions between two virtual characters with distinct mesh topology. In order to transfer the expressions it was necessary to find dense correspondences between the models using volume morphing and a cylindrical projection to apply the deformations by RBF.

In our previous work [Vieira et al. 2012], we defined an architecture of characters reproduction, simulating the transmission of genetic traits of the whole body of different creatures through several generations. In this paper, we extended our previous work [Vieira et al. 2012] by proposing a system that automatically applies facial expressions to reproduced characters. We also defined a parameter to map the genetic information, identifying the origin of each gene from the initial parents. The mesh adaptation system was also optimized, improving the performance of the simulated reproduction, and accelerated the generation of expressions.

3 Adaptation System

The techniques discussed in this section are used both to adapt the meshes in the simulated reproduction and to apply expressions. They are general methods to be used with any triangulated meshes.

3.1 Translation Adaptation

The adaptation of some measurements consists of translating moving spherical influence zones that are positioned and dimensioned to cover areas that require adjustment.

In our method, deformation control vertices are not necessary. Instead, it is the influence sphere that moves, forcing the vertices inside the sphere to adjust their positions. This method works as if the center of the sphere were a vertex of the mesh and its movement changes the position of the influence zone.

The adaptation by translation of the influence zones is illustrated in Figure 2 and is described in equations 1, 2, 3 and 4. The translation is divided into N steps, which are shown in those equations. In each step, the algorithm is applied again to recalculate the new position of the vertices. In Figure 2, two influence zones move in opposite directions to change the width of the nose. The initial configuration is depicted in Figure 2a, and the final configuration, showing the new positions of the affected vertices, is depicted in Figure 2b.

$$\mathbf{x}_i(t + \delta t) = \mathbf{x}_i(t) + \delta \mathbf{v} f_i(t) \quad (1)$$

where

$$f_i(t) = \begin{cases} \frac{R - \|\mathbf{x}_i(t) - \mathbf{x}_c(t)\|}{R} & \text{if } \|\mathbf{x}_i(t) - \mathbf{x}_c(t)\| < R \\ 0 & \text{if } \|\mathbf{x}_i(t) - \mathbf{x}_c(t)\| \geq R \end{cases} \quad (2)$$

$$\mathbf{x}_c(t) = \mathbf{x}_c(0) + t \cdot \mathbf{v} \quad 0 \leq t \leq 1 \quad (3)$$

\mathbf{v} is the translation vector, \mathbf{x}_i is the position of the vertex, \mathbf{x}_c and R are, respectively, the center and the radius of the sphere.

$$\delta \mathbf{v} = \frac{1}{N} \cdot \mathbf{v} \quad (4)$$

The deformation caused by the spherical influence zone in each step of the movement is limited to the vertices of the mesh that are strictly inside the influence zone. The deformation is more pronounced at points that are closer to the center of the sphere and is exactly zero at the edge of the sphere. This type of adjustment generates deformations with smooth transitions between regions inside and outside of the influence zone. The radii of the influence zones

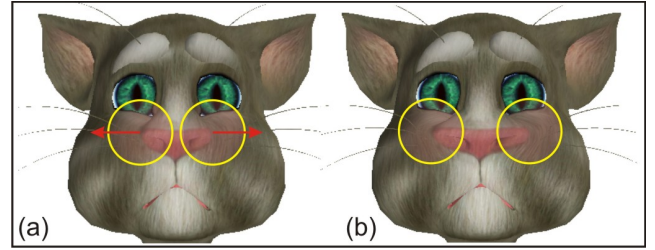


Figure 2: Widening the nose by translation. (a) Before translation. (b) After translation.

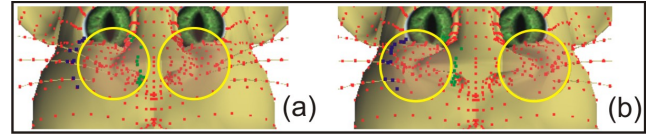


Figure 3: Moving influence zones.

may also vary when it is necessary to affect a larger or a smaller area of the manipulated model.

Note that, as the sphere is translated, some vertices that are close to the border of the influence zone will go outside the sphere while others will go inside (Figure 3). In Figure 3a, the points shown in blue entered the influence zone as it moved to the position indicated in Figure 3b and have their positions adjusted. Similarly, the points highlighted in green went outside the influence zone after the translation of the sphere. This gradually moving influence zone system presents no discontinuity.

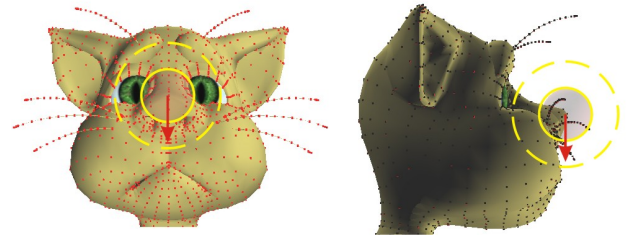


Figure 4: Optimization in the adaptation system.

To reduce the search for the vertices, which lie within the influence sphere, an optimization is defined as illustrated in Figure 4. When the influence sphere begins to move, a second sphere with a radius twice as large as the influence sphere's radius is statically positioned in the original location of the influence sphere. The internal vertices of this second sphere are stored and used as the search set in the translation, eliminating the cost of searching for distant vertices that could never go inside the moving influence sphere. If the influence sphere leaves the field of the static sphere, this larger sphere is instantly repositioned to the current location of the moving sphere. That optimization improves the simulation performance.

3.2 Body Adaptation

In some cases, it is necessary to use a different type of deformation. Measurements of the arms, legs, neck and some other parts use an influence zone with a shape that works more like a cylindrical shape. In those cases, it is necessary to isolate regions so that adjustments are not made in unwanted vertices. Figure 5 illustrates the adaptation of one measurement of the body. To increase the length of the body, we start with the two landmarks, P_0 and P'_0 , that define this measurement (Figure 5a).

With the two landmarks, we find a vector \mathbf{n} that defines the direction of the movement. Next, it is necessary to find the equations of the planes that are perpendicular to \mathbf{n} .

Let \mathbf{r}_0 be the position vector of point P_0 in the plane, and let \mathbf{n} be

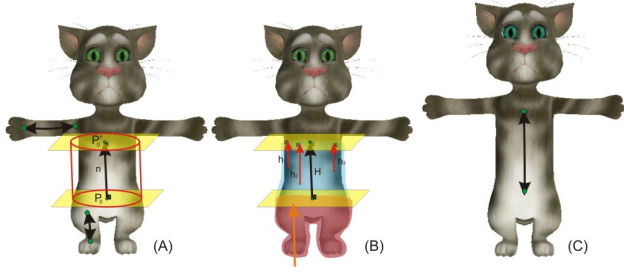


Figure 5: Application of a cylindrical influence zone deformation.

the vector that is normal to the plane. Thus, the equation of the plane is the following:

$$\mathbf{n} \cdot (\mathbf{r} - \mathbf{r}_0) = 0, \quad (5)$$

which can be written in component form as follows:

$$n_x(x - x_0) + n_y(y - y_0) + n_z(z - z_0) = 0. \quad (6)$$

The same procedure is used to find the parallel plane through point P'_0 . With the height of the cylinder and the top and bottom planes, it is possible to find the distance between a vertex and the plane through point P'_0 (Figure 5b) and to compare it with the height of the cylinder, by using Equation 11.

Given a plane $ax + by + cz + d = 0$ and a point $P_1 = (x_1, y_1, z_1)$, the distance from this point to the plane is computed as:

$$D = \frac{|ax_1 + by_1 + cz_1 + d|}{\sqrt{a^2 + b^2 + c^2}}. \quad (7)$$

The points located between the two planes are affected by the motion of point P'_0 (the blue region in Figure 5b). Similar to the influence sphere, the points are adjusted using Equation 1. However, the distances in Equation 2 are replaced with the distance from the point to the plane through P'_0 , and R is replaced with the current distance between the planes through P_0 and P'_0 . The points in the red region are rigidly linked to the plane through P_0 .

Applying the deformation, we obtain the results exemplified in Figure 5c.

4 Reproduction of Virtual Characters

In this section, we summarize the biological process involved in the generation of virtual characters by simulated diploid reproduction. More details can be found in our previous work [Vieira et al. 2012].

4.1 Identification of genetic traits

The first step of the process is to identify a number, n , of phenotypic traits, such as lengths of limbs and measurements of head, eyes, nose mouth, ears and chin; which will be used in the construction of a virtual character. Each of those traits can be seen as information necessary to construct part of the model and will be represented by genes of diploid individuals. In humans, the measurements of the face and body are well defined by anthropometry [Kolar and Salter 1997].

4.2 Storage of the genetic information

The n traits identified in the first step of the process constitute n pairs of allele genes that are distributed across m pairs of homologous chromosomes. Let C be the set of pairs, c_i , of homologous chromosomes,

$$C = \{c_1, c_2, \dots, c_m\}. \quad (8)$$

Each chromosome pair, c_i , consists of two homologous chromosomes: chromosome c_i^M , which originates from the male progenitor; and chromosome c_i^F , which originates from the female progenitor. Thus,

$$c_i = (c_i^M, c_i^F). \quad (9)$$

In our model, the number of genes per chromosome does not have to be the same for all chromosomes. Hence, denoting n_i as the number of genes in chromosome i , the data structures for the homologous chromosomes in c_i are:

$$c_i^M = \{g_{i1}^{M_1}, g_{i2}^{M_2}, \dots, g_{in_i}^{M_i}\} \quad (10)$$

$$c_i^F = \{g_{i1}^{F_1}, g_{i2}^{F_2}, \dots, g_{in_i}^{F_i}\}$$

Note that, in Equation 10, M_k takes the value M if the gene $g_{ik}^{M_k}$ comes from the male progenitor's father and takes the value F if the gene $g_{ik}^{M_k}$ comes from the male progenitor's mother. Likewise, F_k takes the value M if the gene $g_{ik}^{F_k}$ comes from the female progenitor's father and assumes the value F if the gene $g_{ik}^{F_k}$ comes from the female progenitor's mother. This identification parameter makes it possible to do the complete mapping of inherited genes. In the case studies, the provenance of genes of selected characters are shown in order to demonstrate the traceability of genes provided by our model.

4.3 Generation of gametes

Each parental genetic structure is submitted to a simulated meiosis process to generate haploid cells, called gametes. The simulation is focused on the following meiosis' processes: the process of chromosome duplication, and the random processes of crossover, chromosome alignment at metaphase I and chromosome alignment at metaphase II.

In the crossover simulation, each chromosome c_i^M in a pair of duplicated chromosomes $^d c_i (c_i^M c_i^M, c_i^F c_i^F)$ exchanges genes with its homologous chromosome c_i^F .

After crossover, during Metaphase I and II, there are two random processes alignment of chromosomes resulting in four gametes.

The chromosomes are vectors with a group of genes and the weights associated with them. The weights define the degree of dominance of each gene and can be defined in the initial progenitor's configuration or can be randomly assigned during fecundation (this randomization can be performed to ensure more possible combinations). More details can be found in our previous work [Vieira et al. 2012]. The mixture of genes results from meiosis' three random processes of crossover, Metaphase I and Metaphase II. At the end of meiosis, we have four gametes from each parent.

4.4 Fecundation

If only one meiosis simulation is run for each parent, then the male pool of gametes would have four spermatozoids, and the female pool would have four ova. From those small pools, only sixteen distinct children would be generated after fecundation. Fecundation is the union of the two haploid cells, spermatozoid and ovule, to form a diploid cell of the descendant, which has m pairs of homologous chromosomes.

4.5 Implementation of Character Reproduction

To start the simulation, we must establish the basis of the system's configuration. First, we define the model of the character to be considered as the reproductive agent. At this point, we only have a mesh of the model. The next step is to define the set of measurements of the face and body, which will be used as traits in the simulation (Figure 6). Next, we select vertices (landmarks) in the mesh that help define those measurements. This selection is made by a simple picking system. Together with the measurements, we also define the radius of the influence spheres to adapt each measurement in the mesh (that is explained in section 3). Finally, we define the skin and eye colors that are to be used (Figure 7). With all of the traits defined, we construct the character genome, that is, the

number of chromosomes and the distribution of the genes within each chromosome.

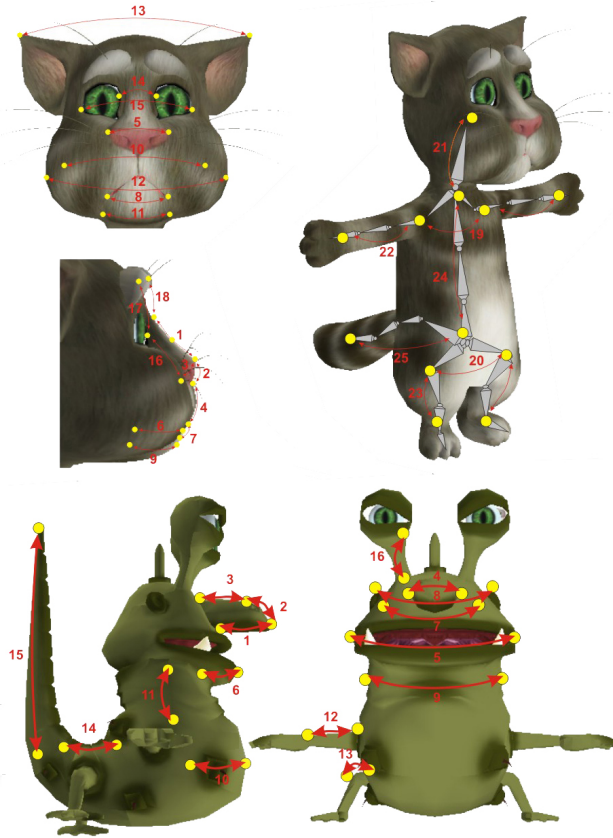


Figure 6: Measurements used in the Cat and Creature simulation.

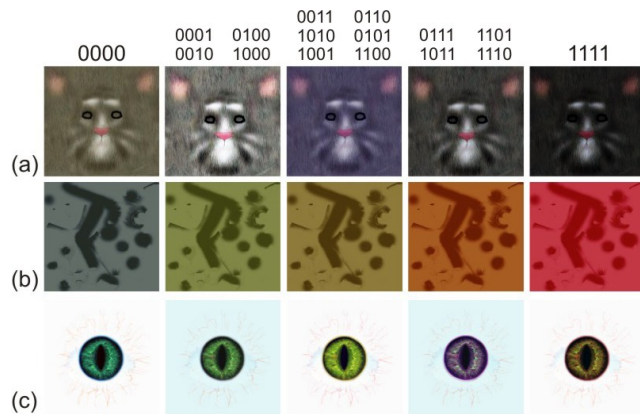


Figure 7: Textures of skin and eye color with possible gene combinations. (a) Skin textures of the Cat model. (b) Skin textures of Creature skin. Eye textures of the Cat and Creature.

With the initial configuration set defined we need a set of initial parents to start the simulation. This process can be accomplished by using professional modeling software or by adapting the measures, using the system that is defined in Section 3, which can be built randomly or by user definition. A key point is that all of the meshes must have the same topology because one of the steps in the simulation (gender adaptation) uses a morphing phase with point-to-point association, and other steps use vertices that are well-defined, to extract and adapt the measurements.

The measures defined in the system’s configuration are extracted from the progenitors’ meshes. This task is performed automatically because the models have the same topology. We simply obtain the positions of the vertices (landmarks), and we compute and save the measurements. Then, we construct the parents’ genomic structures according to the definition used in the last step of the sys-

tem’s configuration. Because the progenitors do not have ancestors, we define them as homozygous, in other words, the pairs of traits in the chromosomes are the same measurements extracted from the meshes.

With all this initial configuration, now it is possible to apply the reproduction simulation. The whole process is summarized in Figure 8. The details of each step will be described in sequence.

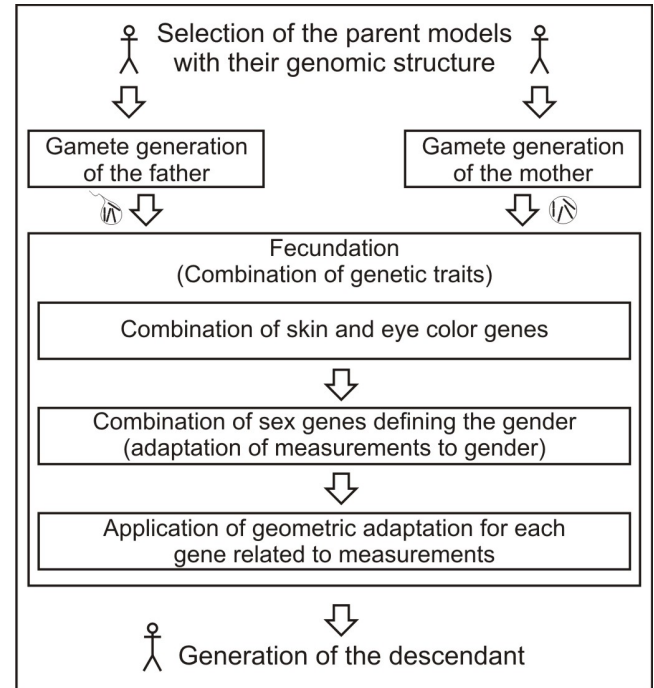


Figure 8: Steps of character reproduction.

Step 1: Selection of the parent models with their genetic structure.

To start the simulation, we select two characters with their genomic structures defined. The process of reproduction will combine, at the end, the characteristics of those two models, generating a child, based on their genetic information.

Step 2: Gamete generation.

This step occurs as explained in Section 4.3. At this point we have the genetic information of the parents that were selected, to be combined and generate the new model.

Step 3: Fecundation.

In this step we combine the genetic information stored in the two gametes that came from the parents. The combination occurs in a sequence where: first, there is a combination of skin and eye color genes; second, there is a definition of sex and a geometric adaptation to the gender of the child; third, it is applied the geometric adaptation for each gene related to measures.

Two genes were used to define the skin color, and other two genes were used to define the eye color. When fecundation occurs, we have four genes for each of these characteristics, two from each parent. Figure 7 shows the possible combinations of dominant and recessive genes of skin and eye color in our case studies.

To accomplish gender transformation, we need to define two meshes that are associated with the creature we are working on, as follows: one mesh represents the average male model, and the other mesh represents the average female model. In the case studies we calculated the average male and female models from the initial progenitors (Figure 9).

When a female descendant is generated, the overall aspect of the geometry must be within anthropometric bounds that are defined for a female. In contrast, when the descendant is a male, the overall geometric aspect has to be within the anthropometric bounds that are

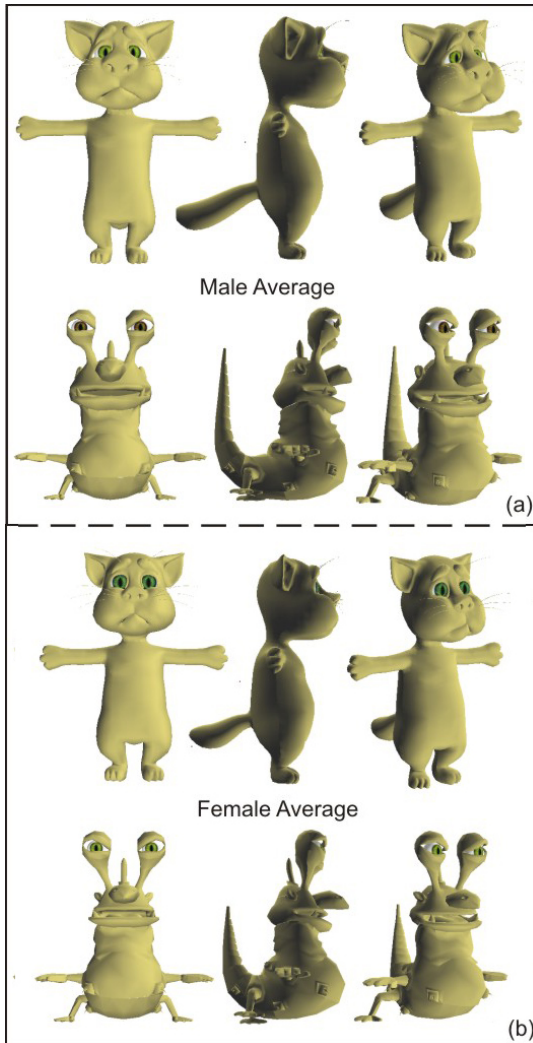


Figure 9: Average male and female models from the initial parents.

determined for males. To generate the mesh of a female descendant, the characteristics that originated from the father are adjusted to female proportions before being combined with the mother's measurements, which are stored in the corresponding genes. In contrast, if the descendant is male, the characteristics that are inherited from the mother are adjusted to male proportions before they are combined with the corresponding characteristics that are inherited from the father. These transformations are performed as described in Figure 10 and are based on Blanz and Vetter's [Blanz and Vetter 1999] work. This process is applied to the mesh and to the measurements that are stored in the chromosomes.

To illustrate the measure adaptation process, we take the gene that determines the width of the nose as an example. The first step, as shown in Figure 9, performs a weighted average of the values that are stored in the allele genes. The weights stored in the genes, together with the corresponding measures, define the type of dominance of the gene. In this case studies, the weights were randomly defined during the measure combination. In complete dominance, the weight of a dominant gene is assigned a value of one, and the value zero is assigned to a recessive gene. The second step consists of applying the combined measure that is obtained in the first step to the morphed model (Figure 9). In the nose width example, we set the influence spheres as shown in Figure 9. The radius of the spheres, for each gene that uses this type of adaptation is defined during the phase of system configuration. With the new measurement calculated, the influence spheres move to the appropriate position. To complete the simulation, those two steps are applied to all of the genes in the genomic data structure.

Step 4: Generation and mapping of genes of the descendants.

Finally, after the previous steps, we have the new reproduced mod-

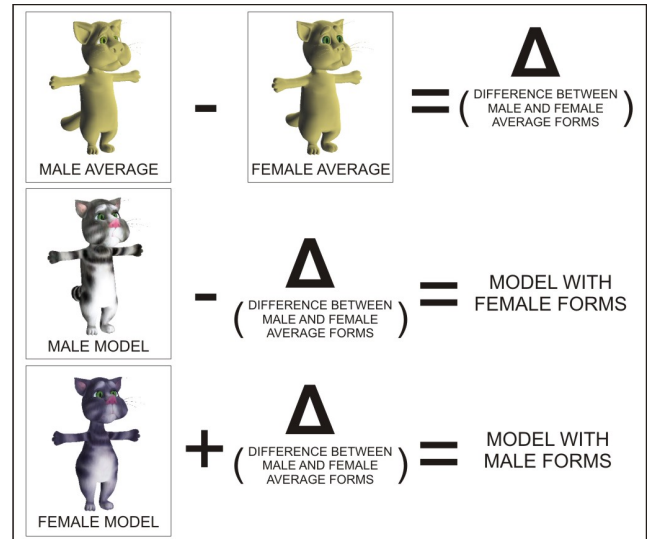


Figure 10: Model adaptation to gender.

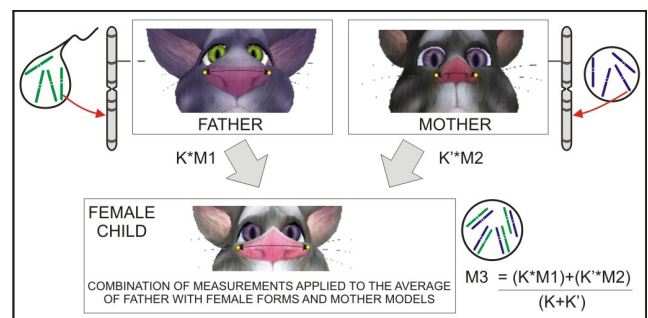


Figure 11: Combination of one measurement of the nose. $M1$ and $M2$ are the measurements from the father and mother together with weights k and k' . The resulting measurement is $M3$.

els. The simulation of the case studies resulted in genealogy trees of character families illustrated in Figures 12 and 13.

With the use of the identification parameter defined in Section 4.2, it is possible to trace the origin of each gene of the genetic structure of the characters generated by the simulation. Table 1 shows, for a given model, the genetic inheritance from the initial progenitors in the Cat case study. One can observe a high concentration of traits inherited from models (a) and (b) compared to the models (c) and (d), in the last generation models (m, o and p). This is because both the father and the mother models have these genes (see the construction of the genealogy tree), which increases the chance of inheritance of them. The model (e) came in the second generation, so it is also more likely to keep their genes in the third generation. Model (i) has 50 % of the genes of the model (e), because this model is an initial progenitor. All initial parents are homozygous, ensuring the transmission of all their measurements (genes) to the next generation.

Model	(a)	(b)	(c)	(d)	(e)
Cat (i)	24.1%	25.9%	0%	0%	50.0%
Cat (j)	24.1%	25.9%	27.6%	22.4%	0%
Cat (l)	31.0%	34.5%	8.6%	12.1%	13.8%
Cat (m)	20.7%	24.1%	13.8%	8.6%	32.8%
Cat (o)	24.1%	24.1%	19.0%	8.6%	24.1%
Cat (p)	22.4%	25.9%	15.5%	6.9%	29.3%

Table 1: Percentage of traits inherited from the initial models in the cat simulation.

For a more detailed analysis, we used the cat model (l) (Figure 14) for a complete mapping of genes, identifying the origin and weights applied to all traits inherited from the initial parents. Table 2 shows the mapping of all genes of model (l), generated through complete

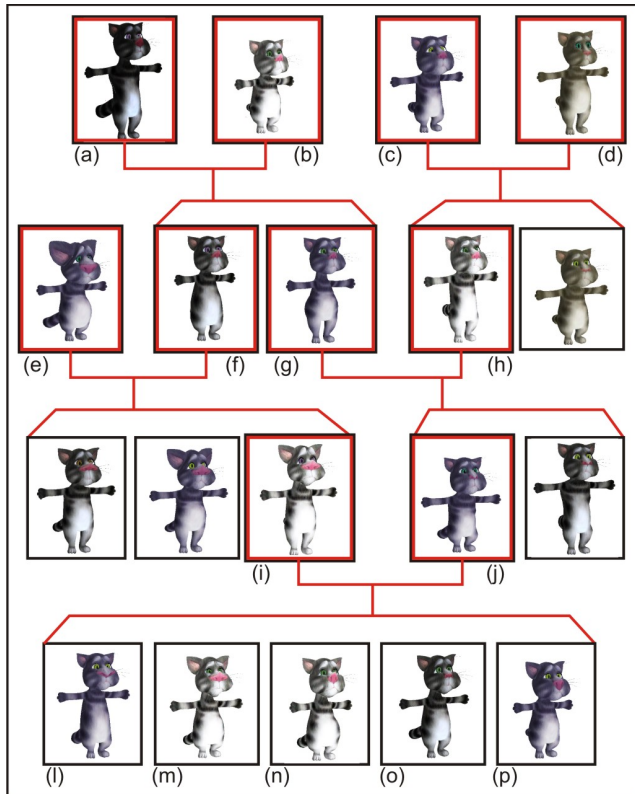


Figure 12: Genealogy tree of Cat models.

Evaluation Measures	Inheritance of model (i)	Inheritance of model (j)
	(a) - (b) - (e)	(a) - (b) - (c) - (d)
Eyes 01	x - 1 - x	x - 0 - x - x
Eyes 02	1 - x - x	x - x - 0 - x
Skin 01	x - 1 - x	1 - x - x - x
Skin 02	x - 0 - x	x - 0 - x - x
Measure 01	x - x - 1	x - x - 0 - x
Measure 02	x - x - 0	x - x - x - 1
Measure 03	0 - x - x	1 - x - x - x
Measure 04	0 - x - x	1 - x - x - x
Measure 05	x - x - 1	x - x - x - 0
Measure 06	x - x - 1	x - 0 - x - x
Measure 07	1 - x - x	x - x - x - 0
Measure 08	x - 1 - x	0 - x - x - x
Measure 09	x - 0 - x	x - 1 - x - x
Measure 10	x - x - 0	x - x - 1 - x
Measure 11	0 - x - x	x - 1 - x - x
Measure 12	0 - x - x	x - x - x - 1
Measure 13	1 - x - x	0 - x - x - x
Measure 14	x - 1 - x	x - 0 - x - x
Measure 15	x - 1 - x	x - x - x - 0
Measure 16	0 - x - x	x - x - 1 - x
Measure 17	x - 1 - x	0 - x - x - x
Measure 18	x - x - 0	x - x - 1 - x
Measure 19	x - 1 - x	x - 0 - x - x
Measure 20	1 - x - x	0 - x - x - x
Measure 21	x - x - 1	x - 0 - x - x
Measure 22	x - x - 1	x - x - 0 - x
Measure 23	0 - x - x	x - 1 - x - x
Measure 24	1 - x - x	x - x - x - 0
Measure 25	x - x - 1	x - x - x - 0

Table 2: Genetic mapping of Cat model (l).

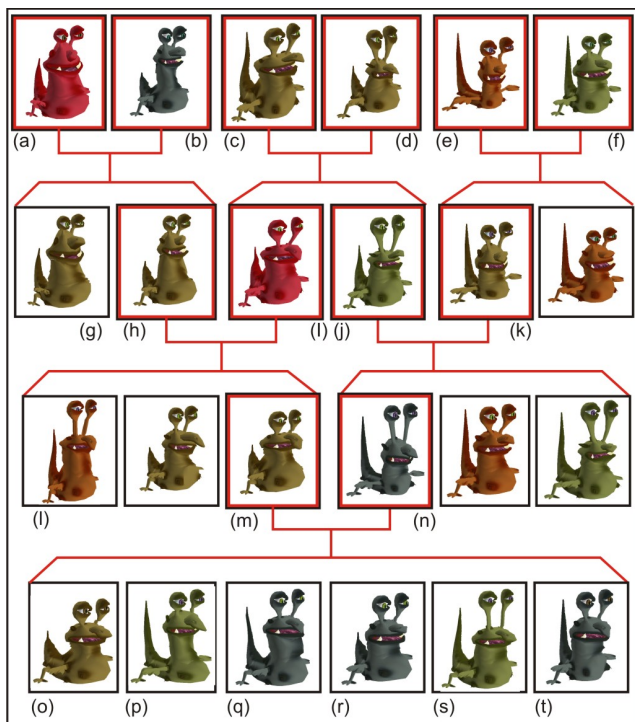


Figure 13: Genealogy tree of Creature models.

dominance, with their respective weights. In this table, traits inherited from each parent are identified with the weight of 1 or 0, which indicate, respectively, whether the gene is dominant or recessive. The genes are combined according to the rule described in Figure 11. In this study, it is possible to identify clearly the action of the dominant genes from the initial progenitors in each generated model. The measures listed in the table are identified and illustrated in Figure 6.

In Table 2, note that model (l) has two dominant and two recessive genes for eye and skin colors. Figure 14 shows that the result is

in agreement with that defined in Figure 7. The genes of the nose width (measure 05) were inherited from models (d) and (e), but the dominant gene (weight 1) was inherited from model (e). The corresponding measures of the models in the first generation (models (a), (b), (c) and (d) in Figure 12) were set markedly different so that one can easily identify them through subsequent generations. Genes that define the height of the legs (measure 23) were inherited from models (a) and (b), but the dominant gene (weight 1) was inherited from model (b). It is possible to check visually that the length of the legs of model (a) is greater than the legs length of model (b) and that the generated model (l) has the leg length identical to model (b). The same analysis can be made for all measurements, comparing the visual result with the mapping in Table 2.

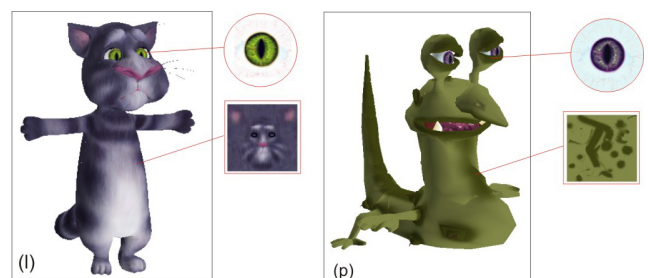


Figure 14: Genetic analysis of reproduced characters.

Table 3 shows, for a given model, the genetic inheritance from the initial progenitors in the creature case study. It is possible to observe a higher concentration of genes inherited from models (c) and (d). This is because both the father and mother models have these genes, which increases the chance of inheriting them. Furthermore, models (a), (b), (e) and (f) are only present in the genetic structure of one of the paternal models.

For a more detailed analysis it was used model (p) (Figure 14) for a complete mapping of genes, identifying the origin and weights applied to all genes inherited from initial parents. Table 4 shows the mapping of all genes of model (p), generated by complete dominance, with their respective weights, as in the previous case study, enabling identification of the dominant measures applied in each of

Model	(a)	(b)	(c)	(d)	(e)	(f)
Model (o)	17,5%	12,5%	20,0%	15,0%	17,5%	17,5%
Model (p)	12,5%	12,5%	37,5%	20,0%	12,5%	5%
Model (q)	15,0%	15,0%	32,5%	10,0%	10,0%	17,5%
Model (r)	15,0%	15,0%	27,5%	15,0%	15,0%	12,5%
Model (s)	10,0%	5,0%	45,0%	20,0%	10,0%	10,0%
Model (t)	10,0%	17,5%	27,5%	22,5%	10,0%	12,5%
Model (u)	7,5%	17,5%	32,5%	22,5%	7,5%	12,5%

Table 3: % genes inherited from the initial models in the simulation of Creature.

the models. The measures listed in the table are identified in Figure 7.

Evaluation Measures	Inheritance of model (i)	Inheritance of model (j)
	(a) - (b) - (c) - (d)	(c) - (d) - (e) - (f)
Eyes 01	1 - x - x - x	x - 1 - x - x
Eyes 02	x - x - x - 0	1 - x - x - x
Skin 01	x - 0 - x - x	0 - x - x - x
Skin 02	x - x - x - 0	1 - x - x - x
Measure 01	x - x - 0 - x	1 - x - x - x
Measure 02	x - 0 - x - x	x - x - 1 - x
Measure 03	x - x - 1 - x	x - 0 - x - x
Measure 04	x - x - 0 - x	x - 1 - x - x
Measure 05	x - x - 0 - x	1 - x - x - x
Measure 06	0 - x - x - x	x - x - 1 - x
Measure 07	x - x - 1 - x	0 - x - x - x
Measure 08	x - 0 - x - x	x - x - 1 - x
Measure 09	x - 0 - x - x	x - x - 1 - x
Measure 10	1 - x - x - x	0 - x - x - x
Measure 11	1 - x - x - x	0 - x - x - x
Measure 12	x - x - x - 0	x - x - x - 1
Measure 13	x - x - 0 - x	1 - x - x - x
Measure 14	x - x - x - 0	x - x - 1 - x
Measure 15	x - 0 - x - x	x - x - x - 1
Measure 16	1 - x - x - x	x - 0 - x - x

Table 4: Genetic mapping of Creature model (p).

In this mapping, one can see in Table 4, that model (p) has three dominant and one recessive gene for eye color and three dominant and one recessive gene for skin color. Figure 14 shows that the result is in agreement with that defined in Figure 7. The genes of the depth of the nose (measure 01) were inherited by his father and mother, from model (c) so that was only possible gene expressed. It is possible to visually see that the only genes that can define the height of the neck of model (p) would be inherited from models (a) or (b). As shown in Table 4, the genes that determine the height of the neck (measure 11) are inherited from models (a) and (c), but the dominant gene (weight 1) was inherited from model (a). The same analysis can be made for all measurements, comparing the result with the mapping of Table 4.

Table 5 shows the performance measurements of the simulation in each case study, with the number of vertices of each model and the time, in milliseconds, to generate one child. In this table we show the performance with and without the optimization introduced in the adaptation system (Figure 4). The reduction of time in the cat study was greater because it has more vertices and the simulation uses more measurements.

Model	Cat	Creature
Number of vertices	9980	4139
Time in milliseconds	640	249
Time in milliseconds with optimization	265	234

Table 5: Time measurements (Pentium Dual Core, 1.46GHz, 2Gb).

5 Application of Expressions

To increase interaction during the game run, we propose a method for automatic application of expressions to the reproduced characters. Facial proportions of a base model is used to transfer expres-

sion to any other model generated in the simulation by the use of translation of spherical influence zones. With this solution, new characters can be generated in the reproduction system and show customized expressions in real time.

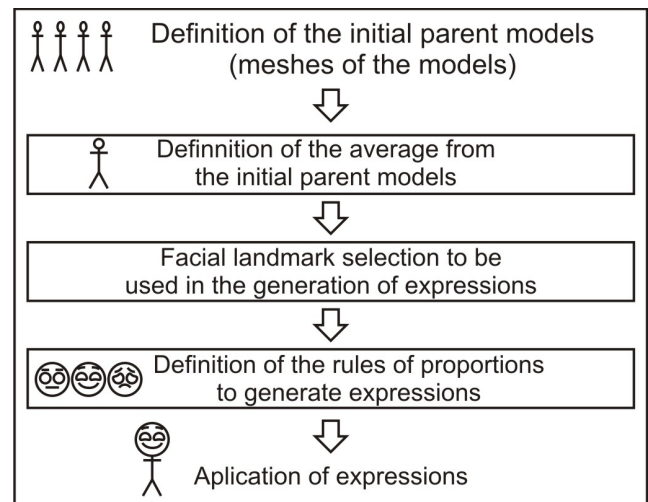


Figure 15: Steps of expressions application.

In this section, the steps of the expressions application (Figure 15) are described by measurements proportions and deformations by motion of spherical influence zones.

Step 1: Definition of the base model.

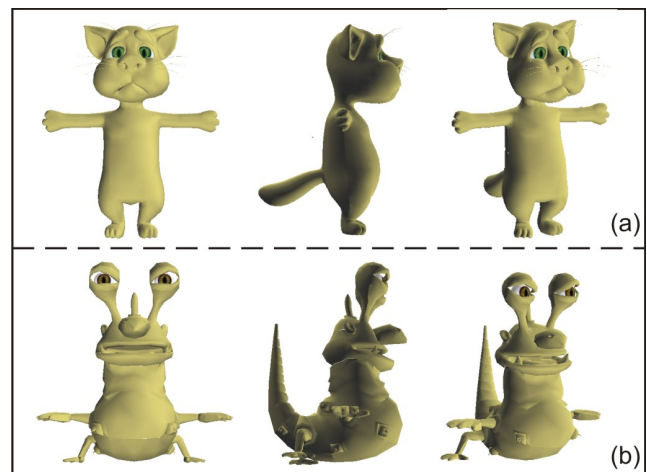


Figure 16: Average Models. (a) Cat. (b) Creature.

Initially, a base model is defined to be used as a template of adaptation rules. To assure that this base model is geometrically close to any model used in the system, the base model is defined as the average model of the initial progenitors of the reproduction system. In the case studies of the cat and creature, the average models are illustrated in Figure 16.

Step 2: Facial landmark selection.

With the average model, the facial landmarks (vertices in the mesh) that will be used are identified to take measurements and generate expressions on the neutral face. This selection is made by a simple picking system. After this, it is necessary to define the desired facial expressions to be transferred and the deformations rules to be applied. Those rules include the direction, proportion of movement and radius of the influence sphere.

Step 3: Definition of rules of proportions to generate expressions.

It is important to mention that the radii of the spherical zones are defined by rules based on distance of points. Models that have larger regions to be moved require larger spheres. Thus, the regions to be

moved are also defined by measures based on landmarks. In Figures 17 and 18, measures k_i are used to define the radius of the spheres in each deformation. After the definition of the landmarks, measures in the neutral face and the spherical radius, it is possible to apply the deformations generating some expressions in the base model, moving the influence zones to the position of each expression (Figures 17 and 18). Measures defined as w_i , in Figures 17 and 18, are used to generate the proportions between the neutral and expression face used in the adaptation. At this point, the proportions of measurements taken from the faces with and without facial expressions are stored to be used later in the other models.

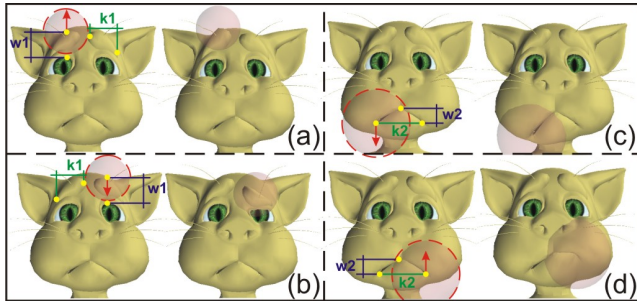


Figure 17: Rules used in the Cat case study.

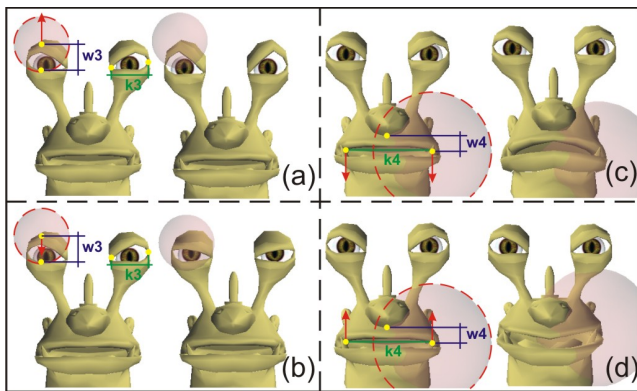


Figure 18: Rules used in the Creature case study.

Step 4: Application of expressions.

Since the models from each case study have the same topology, the landmarks of the models generated by reproduction simulation, are identified automatically. Next, the measures of the neutral faces are used, to generate the expressions in the new models, applying the rules of radius definition and the defined proportions used in the base models (Figures 17 and 18). Expressions are formed translating spherical influence zones centered in landmarks, causing the deformations of the new mesh. Tables 6 and 7 shows information of radius, proportions and final movement of some characters from the cat and creature case studies. Figures 19 and 21 shows the final result of expression application to models of the last generation of the cat and creature genealogy trees (Figures 12 and 13). Figures 20 and 22 shows the graphs of movement of the spherical influence zone in each expressions application to the models (l) to (p) from Cat genealogy tree and models (o) to (u) from the Creature genealogy tree. From the graph of Figure 20 it is possible to see that model (m) had little movements in rules (c) and (d), compared to the other models. In Figure 19 we can see that measure w_2 of model (m) is for sure the shortest one. Table 6 also indicates that the measure w_2 of model (m) is 0.486 shorter than measure w_2 of the average model. The graph of Figure 22 also shows big variations in the movement of the mouth. Table 7 shows that the radius defined to move the mouth of model (p) is much greater ($0.75k_4 * 1.227$) than the radius defined by the average model ($0.75k_4$). The gene that defines the measurement of the mouth is illustrated in Figure 7, as Measure 5. If we look at the complete mapping of model (p), in Table 4, the dominant gene of this measurement came from model (c). In the genealogy tree of Figure 13, it is easy to see that model (c) has one of the largest mouths from the initial parents. Although

model (p) has a larger mouth than the average, its movement was much shorter, as we can see in graph of Figure 22. Table 7 shows that the measure w_4 of this model is 0.855 shorter than the measure from the average model.

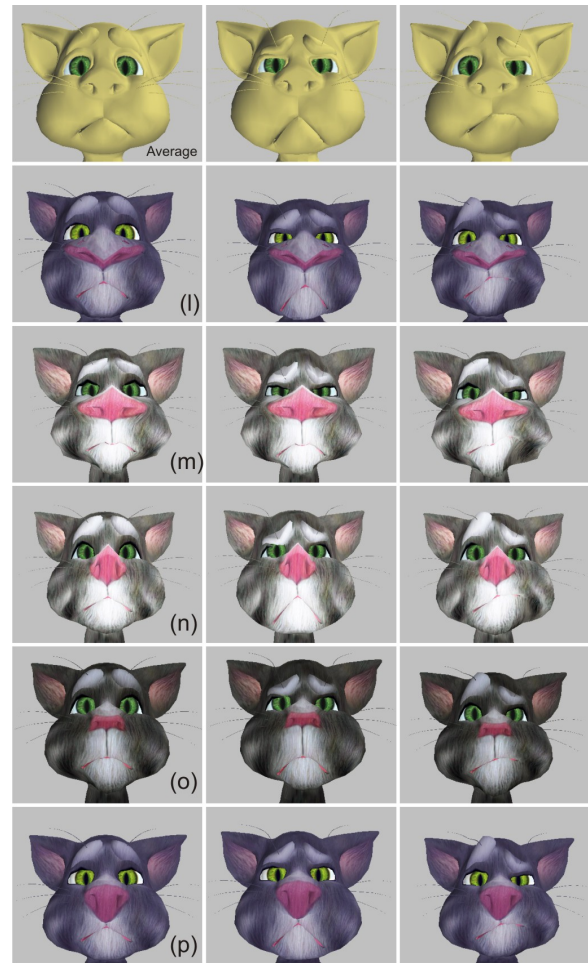


Figure 19: Application of expressions in the cat case study (Models (l) to (p) of Figure 12).

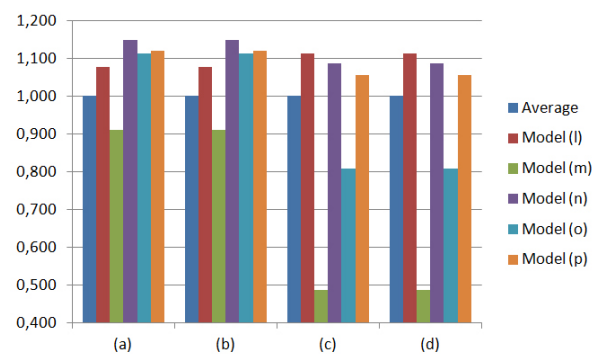


Figure 20: Graph showing the movement of the influence sphere in each Cat case study compared to the average model. (a), (b), (c) and (d) are the rules indicated in Figure 17.

With the proposed system, it is possible to transfer facial expressions in various creatures using the same simple and fast adaptation system used in the reproduction simulation.

6 Conclusion

In this paper we presented a method for facial expressions application to reproduced virtual characters. With the reproduction of virtual characters we can ensure high variability of physical characteristics and transmission through the generations. After the sim-

Model	Rule	Radius	Proportion	Movement
Average	(a)	0.75*k1	1.2	1.2*w1
Average	(b)	0.75*k1	0.6	0.6*w1
Average	(c)	0.75*k2	1.9	1.9*w2
Average	(d)	0.75*k2	-0.15	-0.15*w2
Cat (l)	(a)	0.75*k1*1.02	1.2	1.2*w1*1.078
Cat (l)	(b)	0.75*k1*1.02	0.6	0.6*w1*1.078
Cat (l)	(c)	0.75*k2*1.08	1.9	1.9*w2*1.113
Cat (l)	(d)	0.75*k2*1.08	-0.15	-0.15*w2*1.113
Cat (m)	(a)	0.75*k1*1.02	1.2	1.2*w1*0.910
Cat (m)	(b)	0.75*k1*1.02	0.6	0.6*w1*0.910
Cat (m)	(c)	0.75*k2*1.10	1.9	1.9*w2*0.486
Cat (m)	(d)	0.75*k2*1.10	-0.15	-0.15*w2*0.486
Cat (n)	(a)	0.75*k1*1.02	1.2	1.2*w1*1.148
Cat (n)	(b)	0.75*k1*1.02	0.6	0.6*w1*1.148
Cat (n)	(c)	0.75*k2*1.08	1.9	1.9*w2*1.087
Cat (n)	(d)	0.75*k2*1.08	-0.15	-0.15*w2*1.087
Cat (o)	(a)	0.75*k1*1.02	1.2	1.2*w1*1.112
Cat (o)	(b)	0.75*k1*1.02	0.6	0.6*w1*1.112
Cat (o)	(c)	0.75*k2*1.06	1.9	1.9*w2*0.808
Cat (o)	(d)	0.75*k2*1.06	-0.15	-0.15*w2*0.808
Cat (p)	(a)	0.75*k1*1.02	1.2	1.2*w1*1.119
Cat (p)	(b)	0.75*k1*1.02	0.6	0.6*w1*1.119
Cat (p)	(c)	0.75*k2*1.07	1.9	1.9*w2*1.056
Cat (p)	(d)	0.75*k2*1.07	-0.15	-0.15*w2*1.056

Table 6: Rules applied to some Cat case studies.

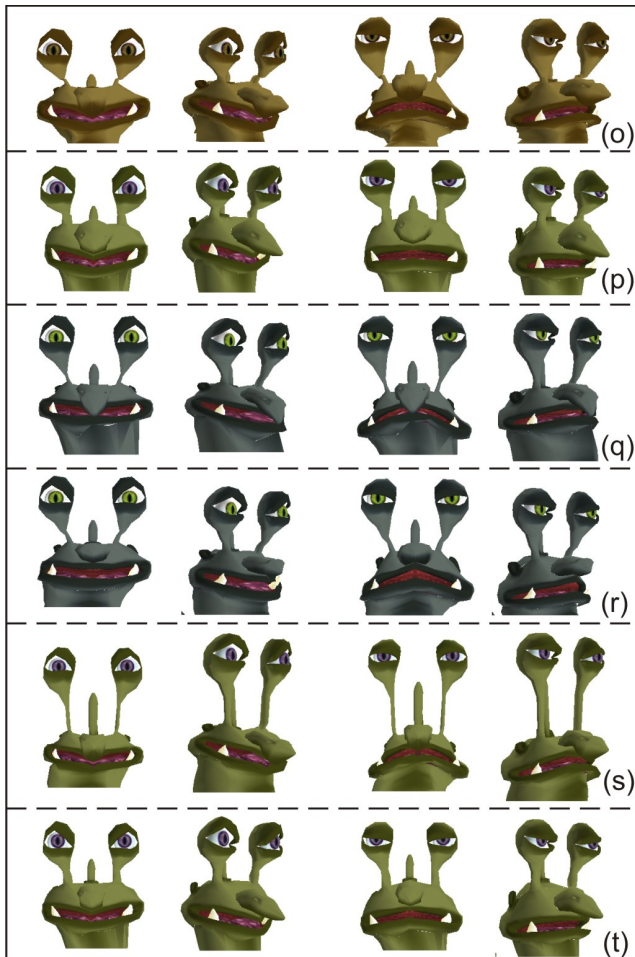


Figure 21: Application of expressions in the Creature case study (Models (o) to (t) of Figure 13).

ulation of reproduction it is possible to increase interaction in the game, by applying facial expressions to the big pool of models generated by the system. The solution is based on the great variability that occurs in diploid reproduction. The process requires an underlying geometric description that fits according to the genetic information stored in artificial chromosomes. The adaptation of the measures is made by the use of deformation spherical and cylindrical influence zones. The proposed adaptation system is intuitive,

Model	Rule	Radius	Proportion	Movement
Average	(a)	0.75*k3	1.4	0.2*w3
Average	(b)	0.75*k3	0.4	0.4*w3
Average	(c)	0.75*k4	0.6	0.6*w4
Average	(d)	0.75*k4	-0.8	-0.8*w4
Model (o)	(a)	0.75*k3*0.992	1.4	0.2*w3*0.951
Model (o)	(b)	0.75*k3*0.992	0.4	0.4*w3*0.951
Model (o)	(c)	0.75*k4*1.383	0.6	0.6*w4*1.157
Model (o)	(d)	0.75*k4*1.383	-0.8	-0.8*w4*1.157
Model (p)	(a)	0.75*k3*0.996	1.4	0.2*w3*0.993
Model (p)	(b)	0.75*k3*0.996	0.4	0.4*w3*0.993
Model (p)	(c)	0.75*k4*1.227	0.6	0.6*w4*0.855
Model (p)	(d)	0.75*k4*1.227	-0.8	-0.8*w4*0.855

Table 7: Rules applied to some Creature case studies.

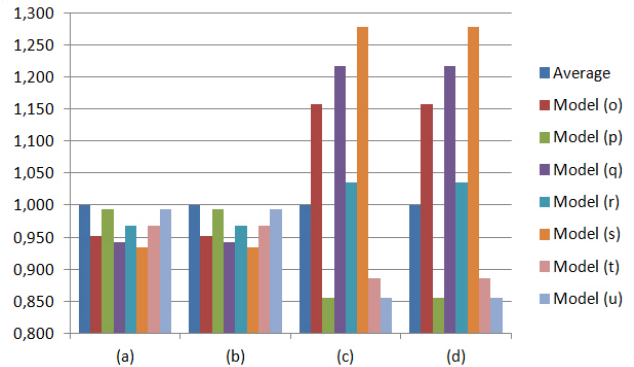


Figure 22: Graph showing the movement of the influence sphere in each Creature case study compared to the average model. (a), (b), (c) and (d) are the rules indicated in Figure 20.

fast and simple, which favors the creation of a generic system that can be applied to various models with different mesh topology and different physical characteristics, as demonstrated in the case studies. With the genetic information of the new models, it is possible to do a complete mapping of the genes, identifying the origin of each gene from the initial parent models. These information can be used in educational and evolution games, and many other applications. To apply the expressions, it is necessary calculate the average model of the initial parents used to generate the new models by reproduction simulation. With the average model, the next step is to identify the types of expressions desired in the game. For each expression, it is necessary to define rules of proportions to generate each expression, using the average model. In this rules it is necessary to define the radius of the influence sphere, based on landmarks, and the proportion of measurements between the neutral face and the face of a given expression. With all the rules defined, the system can apply expressions to the new models generated during the game.

As demonstrated, all processes are generic and can be applied to various models. The solution proposed for the generation of reproduced models, in contrast to those reported in the literature, is versatile and, besides guaranteeing great variability, is capable of generating models that have similar characteristics, simulating a group of the same family or ethnic group, and allows intermix between individuals of different groups, providing specific manipulations such as generation of expressions.

References

ALLEN, B., CURLESS, B., AND POPOVIC, Z. 2003. The space of human body shapes: reconstruction and parameterization from range scans. In *Proceedings of ACM SIGGRAPH, Computer Graphics Annual Conference Series*, 27–31.

BLANZ, V., AND VETTER, T. 1999. A morphable model for the synthesis of 3d faces. In *26th annual conference on Computer graphics and interactive techniques*, 187–194.

DECARLO, D., METAXAS, D., AND STONE, M. 1998. An anthropometric face model using variational techniques. In *Pro-*

ceedings of 25th annual conference on Computer graphics and interactive techniques, 67–74.

- KAHLER, K., HABER, J., YAMAUCHI, H., AND SEIDEL, H. P. 2002. Head shop: generating animated head models with anatomical structure. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*.
- KALRA, P., MANGILI, A., THALMANN, N. M., AND THALMANN, D. 1992. Simulation of facial muscle actions based on rational free form deformations. In *Eurographics*, 59–69.
- KOLAR, J. C., AND SALTER, E. M. 1997. *Craniofacial Anthropometry: Practical Measurement of the Head and Face for Clinical, Surgical and Research Use*. Charles C. Thomas Publisher, Ltd.
- NOH, J., AND NEUMANN, U. 2001. Expression cloning. In *Proceedings of the 28th Annual Conference on Computer Graphics and interactive Techniques SIGGRAPH*, 277–288.
- NOH, J., FIDALEO, D., AND NEUMANN, U. 2000. Animated deformations with radial basis functions. In *VRST*, 166–174.
- PRAUN, E., SWELDENS, W., AND SCHRODER, P. 2001. Physically based grasping control from example. In *Proceedings of ACM SIGGRAPH*, 179–184.
- SEO, H., AND THALMANN, N. 2003. An automatic modeling of human bodies from sizing parameters. In *Proceedings of the symposium on Interactive 3D graphics*, 19–26.
- VIEIRA, R. C. C., VIDAL, C. A., AND CAVALCANTE-NETO, J. B. 2008. A biologically inspired generation of virtual characters. In *Proceedings of the 23rd Annual ACM Symposium on Applied Computing*, 1218–1224.
- VIEIRA, R. C. C., VIDAL, C. A., AND CAVALCANTE NETO, J. B. 2012. Reproducing virtual characters. *Computers & Graphics*, 36, 80–91.