Finger Alphabet by Computer Animation

Jan Miler
j.miler@sh.cvut.cz

Czech Technical University
Department of Computer Science and Engineering
Czech Republic

Abstract

This paper presents some techniques of a human body animation and focuses on a human hand. The animation is demonstrated using a simple application, which interprets text using the international finger alphabet. There were two major problems that have to be sold. The first one is the motion of the hierarchical structure of the skeleton and the second one is the problem of avoiding finger collisions.

Keywords: finger alphabet, manual alphabet, humanoid animation.

1. Introduction

As the 3D technology continues to grow, new and new possible applications are emerging. These applications define new requirements and provide a useful feedback for the 3D graphics community. One of the new requirements is a fast but precise animation of a human body.

This paper is focused on a human body animation and especially on a human hand. The animation is demonstrated using a simple application, which interprets text using the international finger alphabet.

The second section describes one of the possible techniques used when animating a human body [1]. The third section contains a more detailed explanation of a human hand animation such as the hand topology and the concrete methods used. The fourth section is focused on the data structures used for the motion definition. The solution of the problem of the finger collisions is also described there. The fifth section summarizes implementation details of the demo application.

2. Humanoid animation

The most widespread technique of the humanoid animation is based on the bone and knuckle model [1]. Many knuckles of a human body are ignored because the real time animation is required. The performance of a typical computer enables us to augment the existing style of a humanoid animation. The styles of a humanoid animation can be separated by the usage of the knuckles, by the head animation and by the style of the segment representation. The topology of the simple human model contains important knuckles such as a wrist, elbow, shoulder, coxa, knee, ankle and two vertebras, which represent the back and the neck. This model is used in many applications but it cannot suffice the real humanoid representation. The augmentation of the knuckles count to the complete skeleton will make the model more real.

The described model of a skeleton defines the hierarchical structure of the human body but it says nothing about the physiognomy. Thus the muscles and the skin have to be added. Another problem is the simulation of the muscles movement [2]. The muscles can move the bones according to the knuckles and the flexion of a knuckle causes a change in the nearest segments' shapes. Also there is a separated problem with the facial muscles that change the shape of a model without a knuckle flexion.

The described problems can be summarized to two major ones:

- The motion of the hierarchical structure or the skeleton
- The deformation of the body

Next sections are focused on the motion of the hierarchical structure and on the deformation of the body. A detailed description of the hierarchical structure of a hand is in the section 3.1.

2.1 Human Topology

The human body consists of a number of segments (such as the forearm, hand and foot) that are connected to each other by the joints (such as the elbow, wrist and ankle). Topology specifies hierarchical structure of a skeleton. Using the topology we can define relations between joints and segments, which represents knuckles and bones.

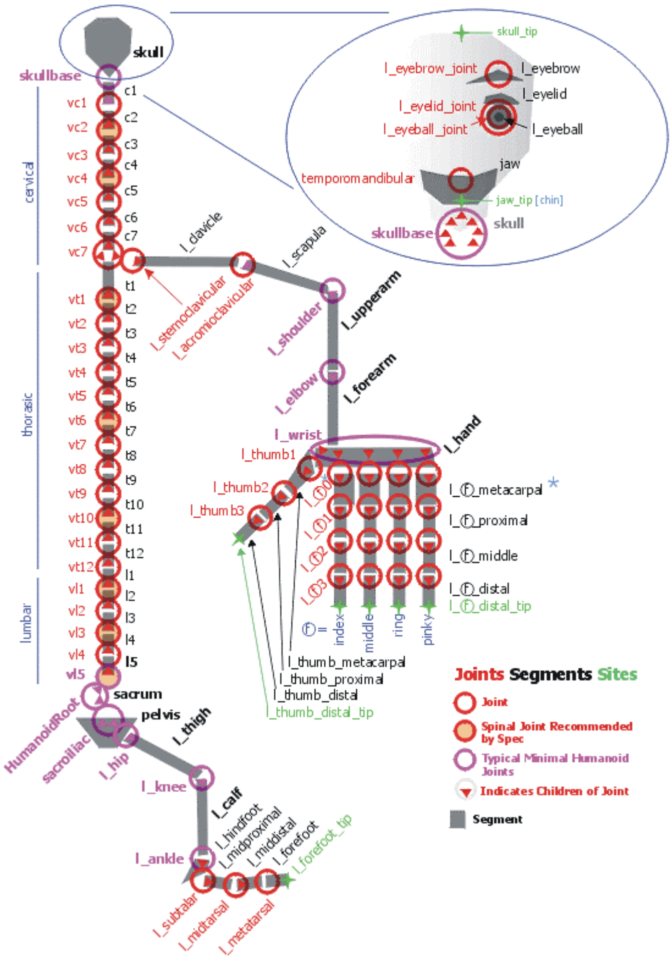


Figure 1: One possible skeletal description used by h-anim group [1]. Humanoid root is in the area of loins.

2.2 Deformation

Deformation of 3D characters and especially human bodies during the animation process is an important but difficult problem. When the body is deformed, the model must be composed from two layers: bones and skin. There are few methods to deform a skin layer.

- Deformations based on implicit surfaces
- Physically-based deformations
- Simple method based on relations between bones and a set of points in the skin layer

For a detailed description see the reference [2].

3. Hand

Human hand has nineteen knuckles. Index, middle, ring and pinky fingers have four knuckles each. Thumb has only three knuckles. Using only fifteen knuckles plus wrist will suffice the purposes of the finger alphabet. The metacarpal knuckles of all fingers without the thumb can be skipped, because their use is very limited. The knuckles between distant and middle segments have relatively the same rotation as the knuckles between the middle and proximal segments.

The mentioned knuckles allow rotation along one axis only. The knuckles between proximal and metacarpal segments allow rotation along two axes. These knuckles, wrist, position of the forearm are the definition of the hand state.

3.1 Topology

The topology of the hand can be represented by a tree where the forearm is the root. Its child is the palm and the palm's children are the proximal segments of the fingers. The distant finger segments are the leaves of the tree.

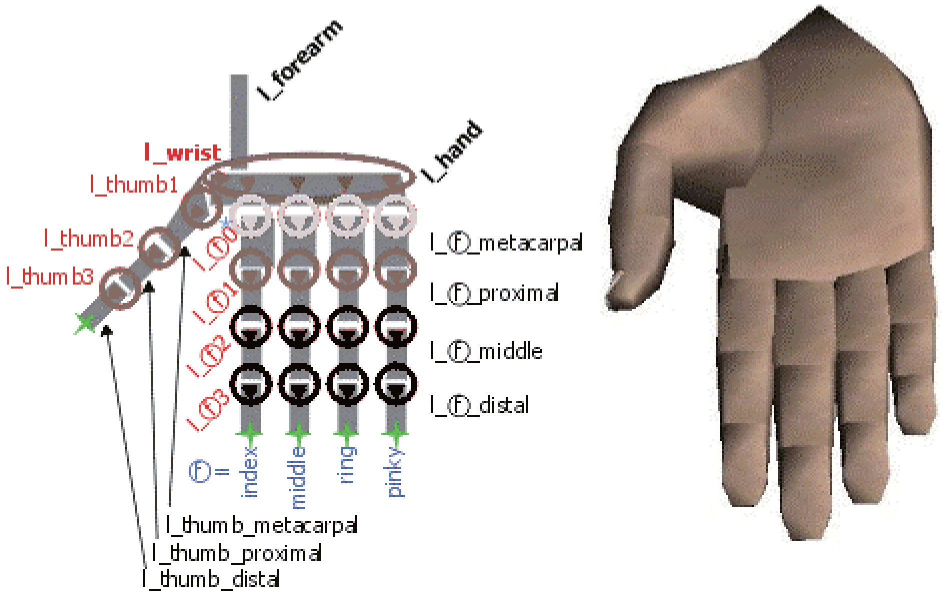


Figure 2: A hand topology. The palm substitutes the light colored joints; gray joints are common joints where each has its own general transformations. Black joints share the rotation in one axis in the application of finger alphabet.

The information carried by each node of the topology is specified in the next section.

3.2 Nodes

Nodes carry two types of information.

- Joint data
  - Rotation
- Segment data
  - Translation
  - Graphical data

The rotation is the transformation of the joint. The translation is the position of the segment relative to its parent. To obtain the absolute position of a given segment, all transformation matrices from the root to the segment have to be multiplied.

For example, transformation of the distant segment of the index finger is computed as follows:

$$M_{id} = M_h \text{ x } R_w \text{ x } T_{ip} \text{ x } R_{ip} \text{ x } T_{im} \text{ x } R_{im} \text{ x } T_{id} \text{ x } R_{id} \text{ x } T_{orig}$$

M – general transformation matrix, T – translation matrix, R – rotation matrix, h – hand, w – wrist, id – distant segment of index, im – middle segment of index, ip – proximal segment of index, T_{orig} – local translation of the segment graphics.

M_h is the transformation matrix, which specifies the absolute position of the hand. Since the root of a hand is the forearm and the wrist, the translation of the palm can be skipped and the wrist rotation is used directly. The next matrices always represent the relative translation and rotation of a finger segment and joint respectively.

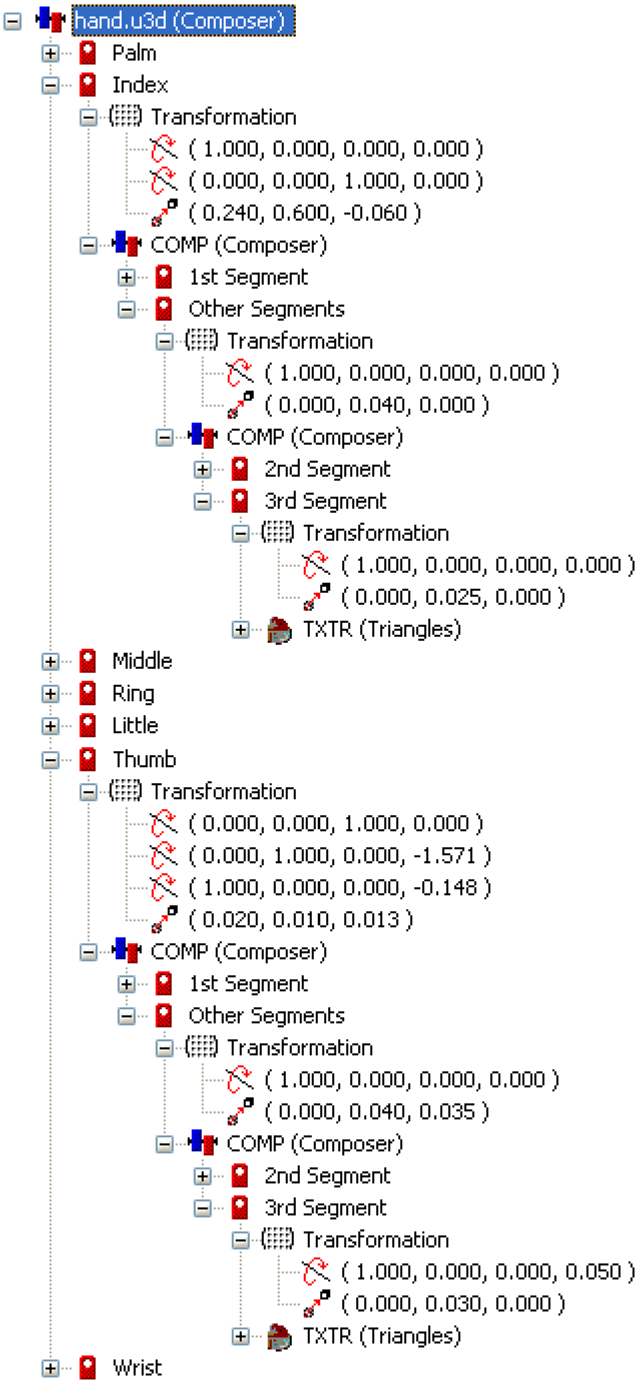
3.3 Graphics

Each node contains also the graphical data. These data specify the shape of the segment and they are made up of closed solid shapes. Intersections of adjacent segments are partially rounded to prevent the cracks when the joint is bowed. Each segment is constructed with the origin in the position of the joint. This approach saves the local translation of the segment graphics. The only transformations are relative translations from the parent joints.

The model of the hand together with its topology was modelled in the Unicorn3D [8] system and the segments were separately stored in the simplified Wavefront (.obj) files. Files in this format consist of a sequence of vertex positions (v), vertex normals (vn), texture coordinates (vt) and blocks of triangles (f). An example of a triangle entry follows:

```
v –0.5 –0.5 0
v 0 0.5 0
v 0.5 –0.5 0
vn 0 1 0
vt 0 1
vt 0.5 0
vt 1 1
usemap skin.jpg
f 1/1/1 2/1/2 3/1/3
```

For complete description see the format specification [7]. The complete model of the hand contains 545 triangles and 321 vertices.



```
hand.u3d (Composer)
    Palm
    Index
        Transformation
            ( 1.000, 0.000, 0.000, 0.000 )
            ( 0.000, 0.000, 1.000, 0.000 )
            ( 0.240, 0.600, -0.060 )
        COMP (Composer)
            1st Segment
            Other Segments
                Transformation
                    ( 1.000, 0.000, 0.000, 0.000 )
                    ( 0.000, 0.040, 0.000 )
                COMP (Composer)
                    2nd Segment
                    3rd Segment
                        Transformation
                            ( 1.000, 0.000, 0.000, 0.000 )
                            ( 0.000, 0.025, 0.000 )
                        TXTR (Triangles)
    Middle
    Ring
    Little
    Thumb
        Transformation
            ( 0.000, 0.000, 1.000, 0.000 )
            ( 0.000, 1.000, 0.000, -1.571 )
            ( 1.000, 0.000, 0.000, -0.148 )
            ( 0.020, 0.010, 0.013 )
        COMP (Composer)
            1st Segment
            Other Segments
                Transformation
                    ( 1.000, 0.000, 0.000, 0.000 )
                    ( 0.000, 0.040, 0.035 )
                COMP (Composer)
                    2nd Segment
                    3rd Segment
                        Transformation
                            ( 1.000, 0.000, 0.000, 0.050 )
                            ( 0.000, 0.030, 0.000 )
                        TXTR (Triangles)
    Wrist
```

Figure 3: The tree notation of topology made up in the Unicorn3D.

4. Motion

Once the model of the hand is complete and its hierarchy is known, the hand moves must be defined. In the finger alphabet application each letter is described by one gesture. The gesture is a sequence of phases. The phase is a static definition of the hand state.

4.1 Phase

The interpolation between two states of the hand is relatively simple. The state is the phase, which define all necessary transformations. Necessary transformations are rotations of all joints and the absolute transformation of the hand. When the phases are interpolated in time the hand moves. The sequence of phases is set up for each letter of the alphabet.

4.2 Gesture

Each letter of the alphabet is expressed by the gesture. The gesture is a sequence of phases. Since phases can be either similar or very different, the time between two phases is computed from the biggest difference of single joint transformations. Therefore the time is related to the difference of phases and various gestures take various time. The gestures for all letters are stored in files. These files are written in XML language [5]. The gesture and the phase are defined in the data type definition language DTD:

```
<!ELEMENT gesture (phase+, phase)>
<!ELEMENT phase
(
    hand_tx, hand_ty,
    hand_y,
    wrist_z, wrist_x,
    thumb_1x, thumb_1y, thumb_1z, thumb_2x,
    index_1z, index_1x, index_2x,
    middle_1z, middle_1x, middle_2x,
    ring_1z, ring_1x, ring_2x,
    pinky_1z, pinky_1x, pinky_2x,
)>
<!ELEMENT “element name” (#PCDATA)>
```

From this definition it is evident that the gesture is always composed from more than one phase. Two important phases are at the end of the gesture. The penultimate phase is the last phase of the letter. The application makes a short pause and then the hand continues through other phases without a pause. The last phase of the gesture is so-called release phase.

4.2.1 Release phase

Since the first phase of a gesture can be very different from the last phase of the previous gesture, segments of the fingers could move through other segments. That is the reason, why the release phase is defined. The release phase is not related to the gesture but it will release fingers from dangerous positions. Moving fingers through themselves is eliminated by this phase but when the release phase is not necessary, it is skipped. There is a release table defined for this purpose. This table specifies when the release phase can be skipped. The release phase can be skipped in about forty percents of cases.

5. Implementation

The Hand Animation program has been implemented in C++ using Microsoft Visual Studio.NET 7.0 Beta2. The program uses the GLUT [3] and OpenGL [4] libraries. The source code was written for both Linux and Windows and has been tested on Windows XP, Windows NT and Linux.

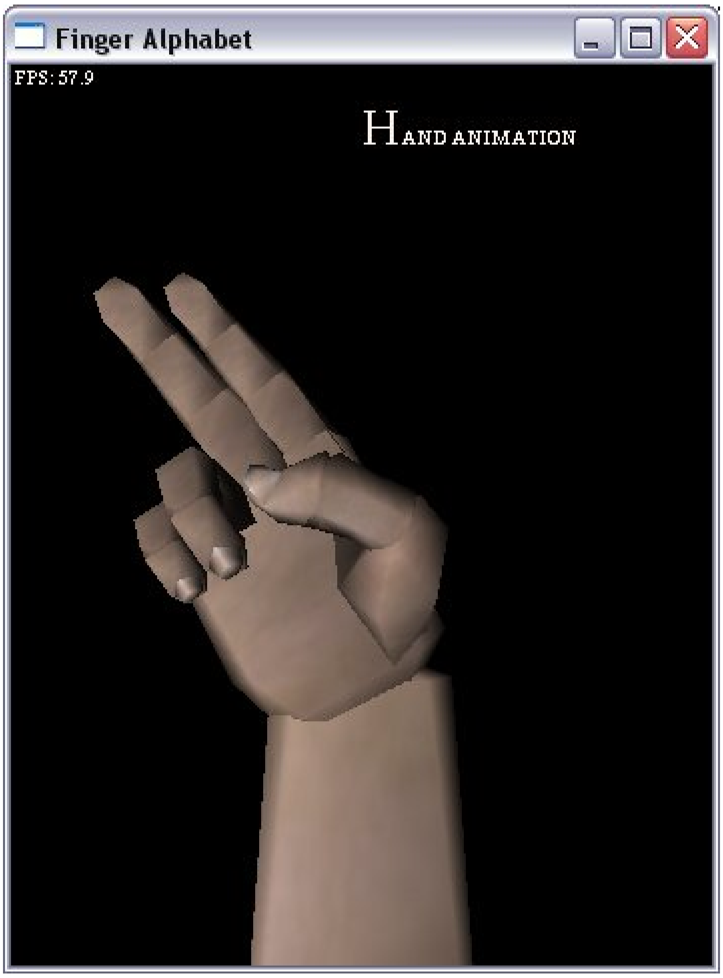


Figure 4: Screenshot of running application.

The application can be switched between full screen and windowed mode. In the upper part of the window a text string with the first letter highlighted is displayed. The hand shows the first letter. Letters can be sent to the application through the keyboard or from the text file specified on the command line. The application enables the rotation of the hand. It can be used for testing the legibility of a finger alphabet from various viewpoints.

5.1 Finger alphabet

International finger alphabet as specified in [6] was used in the application, it is mono-manual and some letters have not only static phases, but they have an additional motion (such as ‘J’ and ‘Z’). Files with gesture definition can be modified to arbitrary mono-manual alphabet.

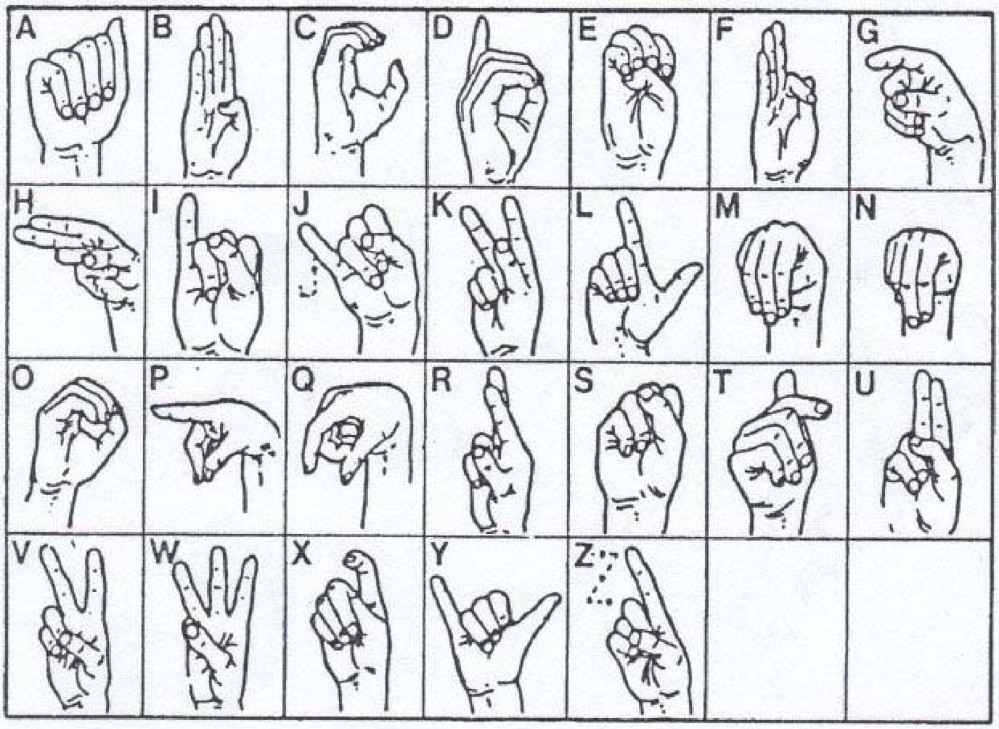


Figure 5: Pattern of the international finger alphabet [6].

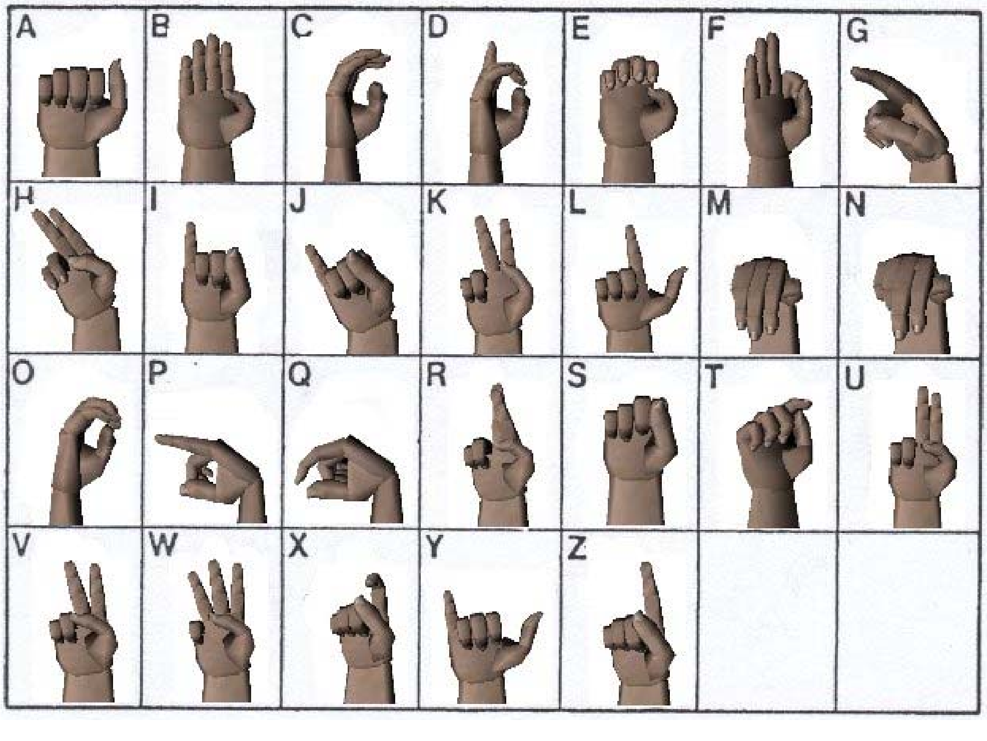


Figure 6: Finger alphabet screenshots.

6. Conclusions and Future Work

This paper has introduced basic principles concerning the animation of the human body and has described a particular application of the hand animation. Software parts of that application can be used separately, for example as hands of a complete human body.

In the future, the application will be used as a base of a library, where the topology of the hand will be modified and where the count of the gestures could be defined. The hand can be rendered specularly as a left hand too. It would allow transformations of letters to any bi-manual alphabet.

7. References

[1] H-Anim Working Group: Humanoid Animation, http://h-anim.org/

[2] The Computer Graphics Lab (LIG): Human Body Deformations, http://ligwww.epfl.ch/

[3] Glut Tutorial, http://www.lighthouse3d.com/opengl/glut/

[4] OpenGL Architecture Review Board.: OpenGL Reference Manual, Addison Wesley Longman, Inc., 1992.

[5] Neil Bradley: XML, kompletni pruvodce, Grada, 2000 [In Czech]

[6] S. Harold Collins, Kathy Kifer: Finger Alphabet Book, Garlic Press, 1992

[7] 3D Data Formats, http://astronomy.swin.edu.au/~pbourke/3dformats/obj/

[8] Unicorn3D description http://cs.felk.cvut.cz/~milerj/u3d/ [In Czech]