

## Virtual Environment Kit for Visual Arts

Jiří Chmelík and Jiří Sochor

Faculty of Informatics

Masaryk University

jchmelik@mail.muni.cz, sochor@fi.muni.cz

### Abstract

We present a virtual environment system for creating various types of visual artistic objects. In this paper we introduce the concept of creating artworks in virtual environment. Our project called “Dynamic Art” comprises several parts: free-hand spatial painting, body motion visualization, and a tool for designing polyhedral sculptures. Each part is briefly described and several results are shown.

### Introduction

Virtual environments (VEs) provide new approaches to many fields of human activity – science, medicine, industry, entertainment and art. Various types of virtual environments can foster entirely new forms of artworks and new approaches to existing art forms. Using VEs, one can create not only paintings or sculptures, but also acoustic, and even haptic, art. They can also join together to create multimedia creations. Artworks created with VEs can be both static, like a sculpture or a composition, and dynamic, like a live performance or an interactive installation where the audience becomes a part of the artwork, influencing it in real-time.

The goal of our research is to explore the possibilities of virtual environments for artistic purposes. We are trying to expand the set of possible results and enrich the creative process by adopting existing artistic concepts to a new – virtual environment.

### Virtual Environment

**Hardware setup.** For input control, any hand-held device with buttons (a mouse, a game controller, etc.) or data gloves can be used. We use a pair of wireless game controllers most often – they provide good ergonomics and a suitable amount of buttons. An optical system based on reflective markers and infrared cameras tracks the position and the orientation of the user and the input devices. This solution provides millimeter accuracy and high tracking speed. Spatial input may be partially simulated by keyboard and mouse, if needed.

For output we use a large-scale stereoscopic projection wall based on polarization. This facilitates the spatial perception of incipient artwork. Stereoscopic imaging by Nvidia 3D vision system and monoscopic rendering are also available. Figure 1(a) illustrates the working environment. The only thing the user must do before using our system is fasten tracking marks (triplets of bright spots in the photo), put on a stereoscopic glasses, and grab the control devices.

With no wires required, the user can move freely in the tracked area of the floor, about  $6m^2$  in area.

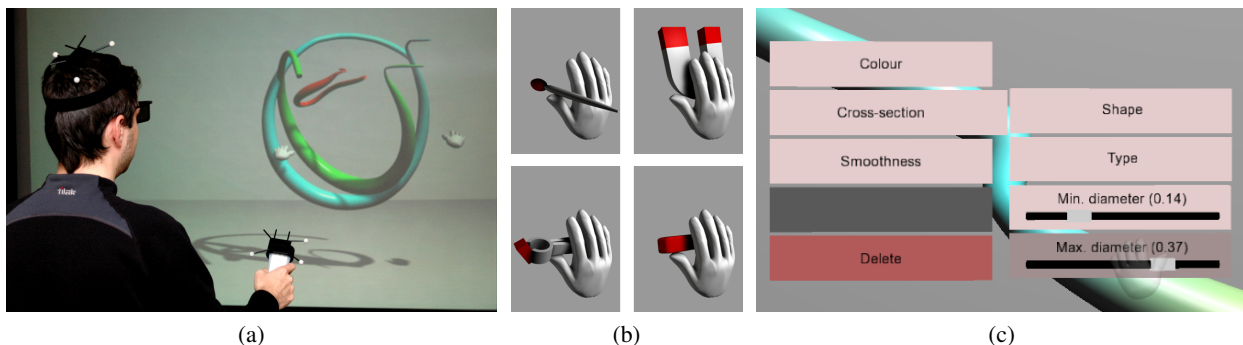
**Software tools.** We are developing a Dynamic Art project as part of our own virtual reality engine VRECKO [8]. This engine provides some core functionality:

- encapsulation of underlying graphics libraries, such as OpenGL and Open Scene Graph<sup>1</sup>;
- communication with special hardware (optical system, controllers, haptic devices, etc.);
- setup of a basic virtual environment (virtual hands, loading models, lighting, etc.).

This functionality and modular architecture of the VRECKO engine simplifies the development of individual VE projects.

We designed our system for users having little or no VE experience. Both the user interface and the virtual tools and hand interaction techniques are clear and easy of use. With their dominant hand, users work with the virtual tools. Movements of a user’s hand are mapped directly onto the movement of virtual hand hold an actual tool. The controller’s main button applies the tool, other buttons are used to adjust parameters of this tool. With their non-dominant hand, users manipulate the artwork by changing its position, orientation and scale. A simple and intuitive “drag and drop” technique is used. The non-dominant hand is also used to control a context menu.

Our graphical user interface includes icons representing tools and main functions, “cursors” of tools, and context menus used to set additional parameters and for system control. See Figure 1 for examples. This interface is fully customizable and its actual layout is slightly different for each experiment of the Dynamic Art project.



**Figure 1** : *Virtual environment: (a) user working with system; (b) cursors of virtual tools: brush, magnet, tongs, and eraser; (c) context menu.*

Using our Dynamic Art project, we have conducted three main experiments to date: spatial free-hand painting, body motion visualization and geometric sculptures design. These experiments are briefly introduced in the following sections.

## Spatial Free-hand Painting

**Related Work.** Several systems for spatial painting exist, built upon different types of virtual environments. The basic idea behind all these systems is the same: provide an extension of the 2D stroke-based creative processes into 3D space. An extensive survey of the interactive 3D drawing area can be found in [6].

CavePainting [7] works in a fully immersive environment and uses physical props (brush, bucket, knobs, etc.) and gestures for system control. A painting is created by layering and arranging individual, virtual 3D brush strokes in space.

In a Surface drawing system [9], a semi-immersive VE (Responsive Workbench) and tangible tools were used. Unlike CavePainting, where each stroke remains individual, adjacent strokes are automatically

<sup>1</sup>Open Scene Graph project: <http://www.openscenegraph.org/>

fused together to form smooth surfaces in the Surface drawing system. Surface drawing also uses more tools for editing artwork: the magnet tool deforms virtual objects, the eraser tool removes surface parts, and the tong tool grabs the virtual object for manipulation of various kinds.

**Brush Strokes.** In our system, the final painting, as in the CavePainting, comprises a number of individual 3D brush strokes. To create a new brush stroke, a user merely presses the button and begins moving their hand. As long as the button remains pressed, the stroke is updated in real-time and follows the path of the hand. For each stroke, users may choose the shape, size and color of the brush.

Strokes are rendered as swept volumes, where both the path and cross-section curves are B-splines. This approach provides high accuracy and thus, strokes are always rendered perfectly smoothly, regardless of the point of view or zoom level in the VE.

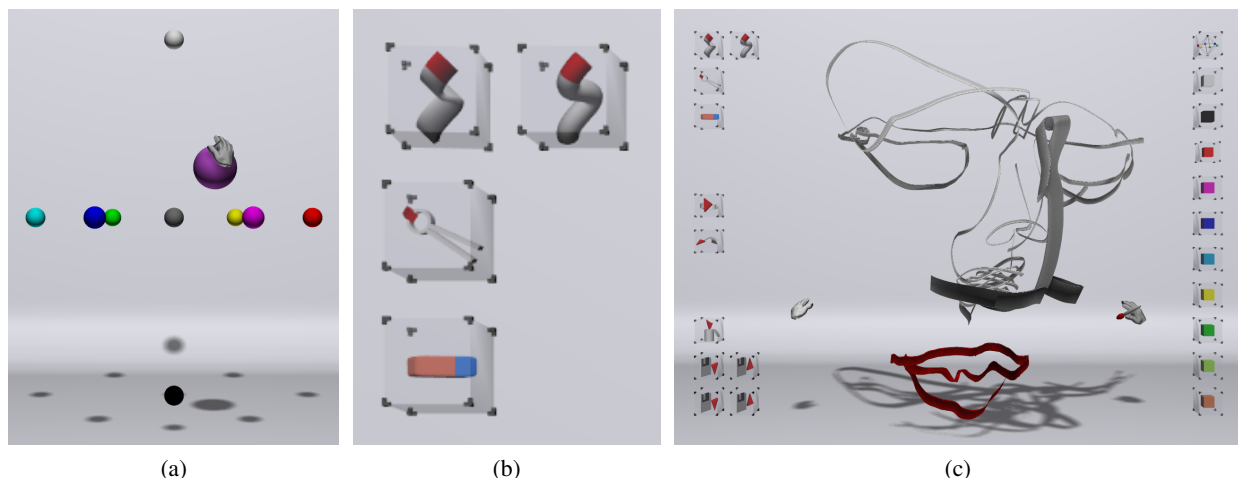
Users may select the brush color directly from a palette of primary colors, or by invoking a 3D HSL color chooser – see Figure 2(a). When this tool is invoked, users can select a color simply by moving their hand through the HSL color space.

**Tools and functions.** Beside brushes for creating new strokes, other tools enable the user to edit the painting: the tongs tools adjusts the position and orientation of a particular stroke and an eraser deletes unwanted strokes. There is also an “Undo” function for reversing the last action.

With their non-dominant hand, users can manipulate the whole painting. After grabbing the painting, users can easily move and rotate it, as if they were holding a real object in their hand.

In the traditional painting process, artists often step back to observe the whole work from a different point of view. Our system achieves this by simply zooming the point of view outwards. To watch a spatial painting from various angles, there is an “Auto-rotation” function. After its activation, the whole artwork begins to rotate at constant speed around its vertical axis, and users can easily check their progress. If the user accidentally “gets lost” in VE, there is a “Home” function – the position, orientation and scale of a painting will be reset to its original state.

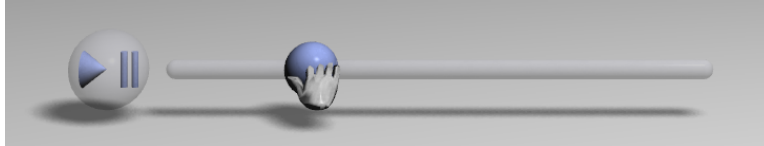
Figure 2(c) shows the layout of the interface. On the left side, there is a group of tools and two groups of functions. Icons on the right side represent the color palette and the spatial color chooser (top icon).



**Figure 2:** Free-hand painting: (a) a spatial color chooser (the bigger sphere represents the selected color); (b) icons of tools: brushes (flat and round), manipulator and eraser; (c) an overall view.

**Presenting artworks.** Due to the spatial nature of created virtual paintings it is natural to present them using the virtual environment. Our application includes an “Audience mode” for this purpose. In this mode, all editing tools and functions are disabled and their icons are hidden, but the painting still may be manipulated by the techniques described above. Thus, the audience member can interact with the artwork using one of the controllers, see it from different angles, and explore various parts of the work by zooming virtually closer to them.

Additionally, we can display the entire life cycle of a virtual painting’s creation using a “Time-line” tool (Figure 3). Using the “Play / pause” icon, the process of a painting’s creation may be re-played stroke by stroke. By dragging the slider, the user can quickly display the entire history of a work-in-progress.



**Figure 3:** *Free-hand Painting: a Time-line tool.*

The finished spatial painting may also be exported as a 3D polygon mesh with high precision, for off-line rendering purposes. Figure 6 (top right) shows an example of a painting created in our system and rendered with external software Cinema 4D™.

### Body Motion Visualization

The process of painting in VE – using whole body movements to interact with a computer – was often described by users as a dance-like motion. To express the abstract aesthetics of motion itself, rather than a figurative painting, we have created a system for the visualization of human body movements. The motion of a user is captured by simultaneously tracking several body parts (head and limbs, usually). For each tracked part a so-called “motion trail” is rendered in real-time. After capturing the desired motion, a user can further customize the visualization of individual motion trails.

The basic concept of a motion trail is derived from the brush strokes described above – one trail represents the movement of one part of the user’s body over a certain period of time. However, the motion trail differs from brush stroke in several features.

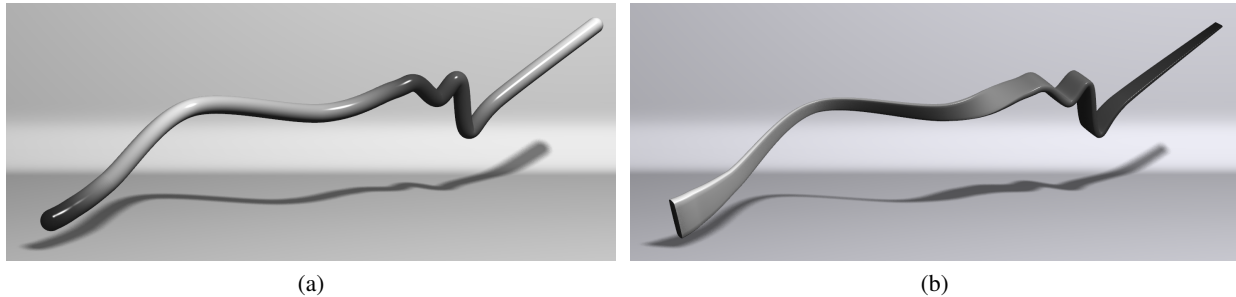
**Characteristics.** To create a comprehensible visualization of motion, several geometrical characteristics are computed for each control point of each trail:

- local coordinate system, curvature and torsion;
- arc length and speed of movement;
- priority.

High accuracy of the motion tracking system may cause an unwanted jitter in the sampled data, e.g., when the tracking marks are not fastened properly on the user or when the user’s hands are shaking. To eliminate this jitter and to render smoother trails, we have implemented a curve optimization algorithm based on repeated removal of points with the lowest priority.

**Visualization.** Similarly to brush strokes, rendering of motion trails is based on swept volumes. To compute the orientation of cross-section of swept volume, we use the Frenet-Serret Frame [4]. Thus, we can generate an optimized tube-like shape around the motion path curve.

We have enriched the visualization by gradients and variation in cross-section shape to distinguish between individual trails and to make visible the movement direction and speed of body parts. See Figure 4.



**Figure 4 :** *The motion trail rendered with: (a) constant diameter and a speed-based color gradient (black is slow, white is fast); (b) speed-based diameter (the faster motion, the smaller diameter) and start-end color gradient (white at start, black at the end).*

In a virtual environment, the audience can interact with the visualization and see motion trails from an arbitrary point of view. The time-line tool is also available, so the dynamics of a motion can be studied. Examples of motion trail visualization are shown in Figure 6 (middle row). More details about this experiment can be found in [3].

## Polyhedra Sculptures

**Related Work.** There already exists a rich history of mathematical sculptures and computer based approaches for design of polyhedral sculptures, e.g. the works of Séquin [10] or Akleman [1]. We have been inspired by work of G. Hart and his polyhedral sculptures made from planar components [5].

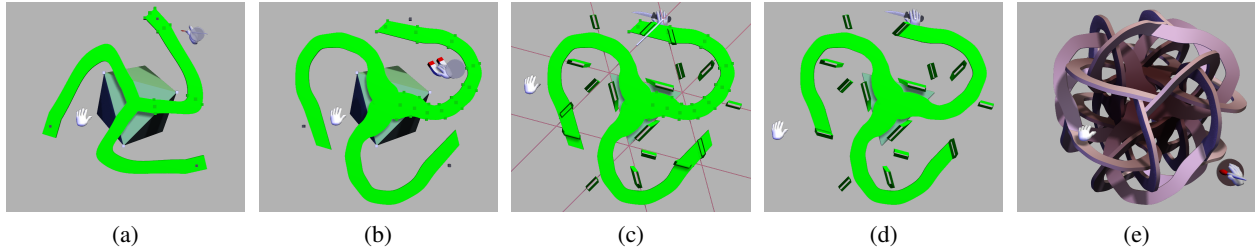
Despite the fact, that many specialized CAD applications are available for mathematical sculpture design, and several CAD systems exist that run in virtual environments, as far as we know, there is no application focused on the design of geometrical sculptures directly in a VE.

**Polyhedra Based Sculptures.** Our experiment is focused on sculptures based on uniform polyhedra. Currently, users can choose one of seven polyhedra, from tetrahedron to rhombic triacontahedron. The extended faces of a such polyhedron form a set of intersecting planes. These planes define locations of sculpture components; each component lies on one face-plane of polyhedron. The component itself is a flat spatial object with a given thickness. The resulting artworks are complex, spatial shapes with intricately interwoven components and with a high degree of symmetry. The goal of this experiment is to exploit advantages of VE to simplify the perception of these complicated shapes and to make the create process more intuitive.

**Component.** At the beginning of the designing process, the shape of each component resembles a simple ribbon. The user can freely extend or bend it using the creator and the magnet tools. New branches can be added to the component by creator tool – smooth connection to the nearest existing branch is computed automatically. The knife tool cuts off parts of the component. To achieve maximum accuracy, the component is stored as set of B-splines.

**Interactions & Visualization.** During editing process, users can toggle between viewing a single component, or a “slice” through the assembly or the entire sculpture. The first two options are useful when the sculpture is complex and its components are heavily intertwined. In the slice mode, only a narrow band beginning just below the component and ending just above it is rendered. Thus, users can clearly see all the intersections of components.

For a better perception of the polyhedron symmetries, intersections of polyhedral planes can be shown as lines (creating a stellations) or as points. Figure 5 depicts various phases of a simple sculpture design and the usage of the mentioned auxiliary elements.



**Figure 5:** *Sculpture design based on an octahedron: (a) A draft of a component created with the creator tool. Intersection points displayed. (b) Creator and magnet tools adjust the shape. (c) Slice mode and stellation pattern turned on to visualize intersections; Knife tool used to cut off the overlapping part. (d) Final shape of the component achieved, auxiliary elements turned off. (e) All components rendered. Brush tool used for coloring the sculpture.*

Besides the above virtual tools, a radial menu lets the user change visibility of sculpture parts and auxiliary elements, and for system control tasks (auto-rotation, undo, save, load, etc.).

As in previous experiments, the final sculpture may be presented in VE or exported for further processing. One component or the whole sculpture may be saved as watertight 2-manifold polygon mesh with arbitrary precision. This allows us to use rapid prototyping devices such as 3D printers, to create physical realizations of designed sculptures (Figure 6, button row). For more details about this experiment, see [2].

## Conclusion and Future Work

We have built a system that supports the creation of various forms of visual artwork directly in a virtual environment. Preliminary testing showed that the user interface is easy to understand for users without previous experience with VE. The real-time spatial input makes interaction with the artwork intuitive and simplifies the creative process. The large-scale stereoscopic output greatly improves the perception of the depth of rendered artworks.

Each of the described experiments can be used in an interactive art process, resulting in interesting pieces of visual art. The utilization of a virtual environment permits new approaches, new visualization methods, and a better work-flow for this artistic process.

**Results.** We have informally evaluated this work through on-site demonstrations with groups of high school and undergraduate university students. We also did several small user studies with established artists. The response was generally very enthusiastic, especially among the artists. Most users were impressed by the possibility of direct spatial interaction with virtual art pieces and by the real-time stereoscopic rendering.

Figure 6 shows several artworks created in our virtual environment. Images labeled as “screenshots” were captured directly from our application. Images labeled as “external renderings” were created using Cinema 4D™ software. In all images depicting motion visualization, the same color scheme was used: orange for the head, green for the right limbs, and blue for the left limbs. Several pictures and videos presenting artworks created in our system are available at the author’s web page<sup>2</sup>.

**Future Work.** The experiments described in this paper are not closed. In Free-hand painting, the audience

<sup>2</sup><http://www.fi.muni.cz/~xchmeli1/research/VEKVA/results>

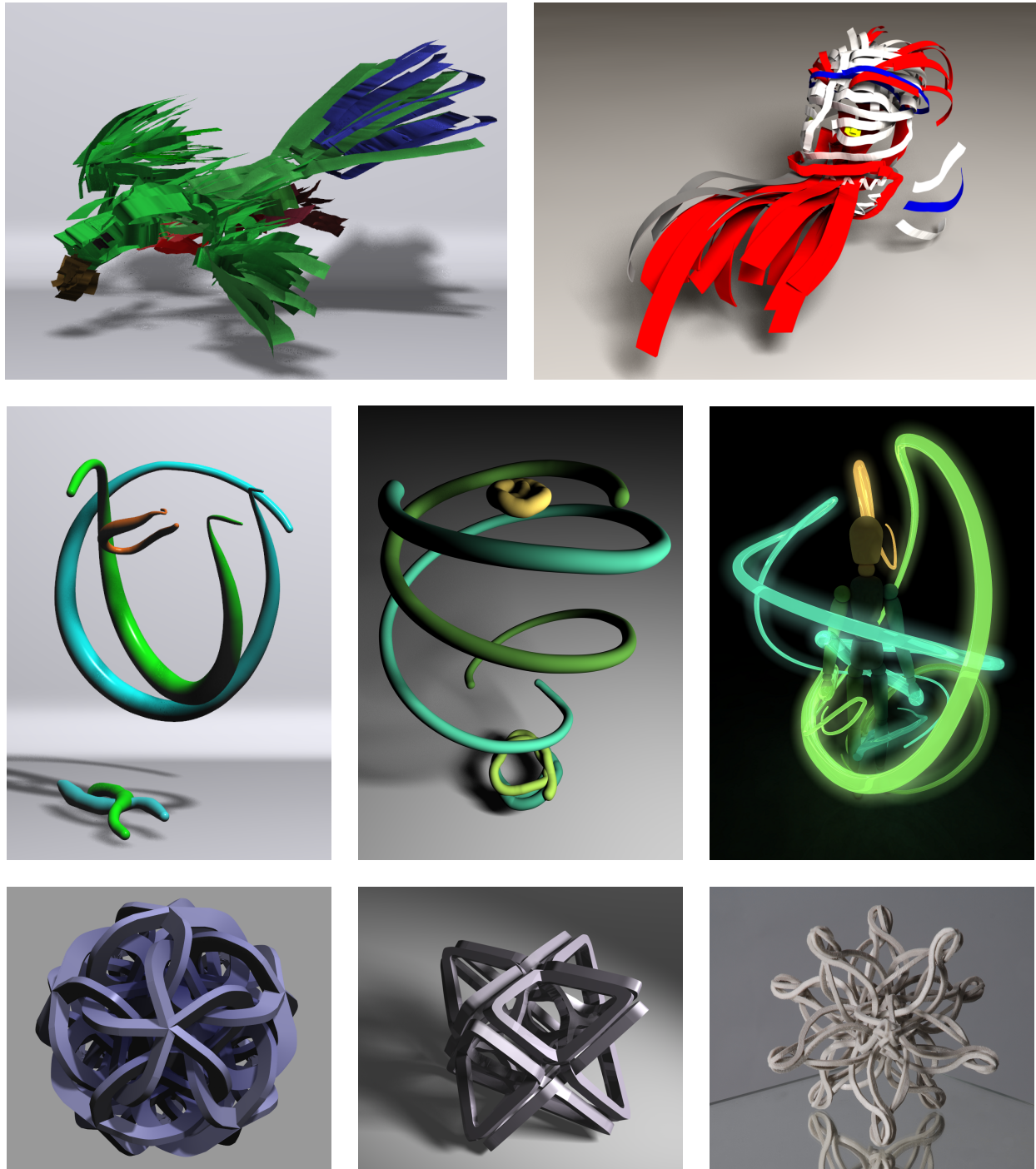
mode will be enriched by an option to “see through the author’s eyes”. The motion visualization system should contain a digital avatar, moving according to created motion trails. In a geometrical sculptures experiment, the set of virtual tools should be improved to allow new types of modifications for the base component. Currently we are implementing a tool for exploring spatial fractal sets and chaotic attractors in VE.

The long-term goal of our research is to bring new experiments to life and find new benefits in the use of virtual environment in other areas of math and arts.

**Acknowledgement.** This work was supported by the Ministry of Education, Youth and Sports of the Czech Republic under the research program LC06008 (Center for Computer Graphics).

## References

- [1] Ergun Akleman, Ozan Ozener, and Cem Yuksel. On a family of symmetric, connected and high genus sculptures. In *Proceedings of the 2006 Bridges Conference on Mathematical Connections in Art, Music, and Science*, pages 145–150, 2006.
- [2] Jiří Chmelík and Jiří Sochor. Virtual environment for designing geometrical sculptures. In *Sigrad '09, The Annual Sigrad Conference, Special Theme: Visualization and Design*, pages 11–18, Goteborg, Sweden, 2009.
- [3] Jiří Chmelík and Jiří Sochor. Body motion visualization in virtual environment. In *Proceedings of the 30TH CONFERENCE ON GEOMETRY AND GRAPHICS*, pages 119–124, Prague, 2010.
- [4] Andrew J. Hanson. Quaternion frenet frames: Making optimal tubes and ribbons from curves. Technical report, Computer Science Department, Indiana University, Bloomington, IN 47405, 1993.
- [5] George W. Hart. Sculpture from symmetrically arranged planar components. In *Meeting Alhambra. ISAMA-BRIDGES 2003 Conference*, pages 315–322, 2003.
- [6] Daniel F. Keefe. *Interactive 3d drawing for free-form modeling in scientific visualization and art: tools, methodologies, and theoretical foundations*. PhD thesis, Brown University, Providence, RI, USA, 2007.
- [7] Daniel F. Keefe, Daniel Acevedo Feliz, Tomer Moscovich, David H. Laidlaw, and Joseph J. LaViola, Jr. Cavepainting: a fully immersive 3d artistic medium and interactive experience. In *I3D '01: Proceedings of the 2001 symposium on Interactive 3D graphics*, pages 85–93, New York, NY, USA, 2001. ACM.
- [8] Vít Kovalčík, Jan Flasar, and Jiří Sochor. Extensible approach to the virtual worlds editing. In *AFRIGRAPH '07: Proceedings of the 5th international conference on Computer graphics, virtual reality, visualisation and interaction in Africa*, pages 31–37, New York, NY, USA, 2007. ACM.
- [9] Steven Schkolne, Michael Pruett, and Peter Schröder. Surface drawing: creating organic 3d shapes with the hand and tangible tools. In *CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 261–268, New York, NY, USA, 2001. ACM.
- [10] Carlo H. Séquin. Computer-aided design and realization of geometrical sculptures. *Computer-Aided Design & Applications*, 4(5):671–681, 2007.



**Figure 6:** Results: Top row: Free-hand Painting – screenshot and external rendering; Middle row: Motion visualization – screenshot and two external renderings; Bottom row: Geometric sculptures – screenshot, external rendering and photo of FDM 3D print.