Space-Time Blending for Heterogeneous Objects

A. Tereshin [1], E. Anderson[1], A.Pasko[2,1], and V. Adzhiev[1]

[1]National Centre for Computer Animation, Bournemouth University, Poole, UK
[2]Skolkovo Institute of Science and Technology, Russia

Abstract

Space-time blending (STB) is an established technique allowing to implement a metamorphosis operation between geometric shapes. In this paper we significantly extend the STB method to make it possible to deal with heterogeneous objects, which are volumetric objects with attributes representing their physical properties. The STB method, used for geometry transformation, is naturally combined with space-time transfinite interpolation, used for attribute (e.g. colour) transformation. Geometry and attribute transformations are interconnected and happen simultaneously in an higher dimensional specific STB space. We use hybrid function representation, unifying function representation with signed distance fields and with adaptively sampled distance fields. We show how the new method works by applying it to 4D animated Cubism.

CCS Concepts: Shape modelling ⟶ *Volumetric models;*

1. Introduction

Space-time blending (STB) [PPK04] is a geometric operation of bounded blending performed in a higher-dimensional space. It realises a metamorphosis operation between shapes of different topology without necessarily establishing their alignment or correspondence. In this work, we extend this method to make it possible to deal with heterogeneous objects in which geometric shapes are considered in concert with their internal attributes, defined for each point of the shape, which can be represented as physically based materials, density, colour and others. The method is supposed to take advantage of keeping all conceptual features of the STB geometric technique, extended to the attributes. The attribute transformation is realised using space-time transfinite interpolation (STTI) [SFA*15], happening in the same conceptual space as STB.

More specifically, in this paper we will concentrate on dealing with such an important attribute as colour. Technically, this means that a higher-dimensional specific STB space in which the geometry transformation is executed is associated with a 4D colour space with an added time dimension. In distinction with [TAF*20] this will make it possible to interpolate dynamic textures simultaneously with ongoing geometry transformation before returning to the initial space thus effectively resulting in a novel unified method for generating a smooth transition between 2D/3D shapes with 2D/3D textures without establishing any correspondences between initial and target heterogeneous objects and their attributes. This method assumes that we use a hybrid function representation (HFRep) with a distance property also extended comparatively with its version presented in [TAFP19].

Most known solutions for volumes or scalar distance field (DF) representations work only with geometry and only few can handle both geometry and attribute transformations. Let us mention only the most representative related works here.

Breen and Whitaker [BW01] introduced a partial differential equation (PDE) based level-set method for representing the deformable surface of a densely sampled scalar function. Colour interpolation is implemented using a trilinear interpolation combined with the scan-conversion closest point method where, unfortunately, the trilinear interpolation only handles simple colour transformations. Dinh et al. [DYTT05] introduced a PDE based method that could transfer textures during shape transformation. Their method is based on solving the Laplace equation for defining flow lines that execute the bijective transformation between input and target objects. For texture mapping they used the Laplace equation with the Laplace-Beltrami operator to establish pointwise correspondences between two objects defined by implicit functions raised into 4D space.

Weng et. al. [WCX*13] introduced a metamorphosis method for objects that are defined using DFs that exploits a PDE-based approach called optimal mass transport (OMT) that provides the optimal way for moving a mass distribution from one domain to another with minimal transportation costs. Unfortunately this method is quite sensitive and cannot handle colour or texture transformations. Solomon et al. [SdGP*15] described a method for solving the OMT optimisation problem using convolutional Wasserstein distances approximated using entropic regularisation. This method can handle interpolation between blocks of pure colour and can be computed independently from the geometry.

PDE based methods are in most cases computationally expensive. While there are methods that are less expensive, these usually have less satisfactory results. Barbier et al. [BGA05] introduced a

DOI: 10.2312/egs.20201014

metamorphosis operation for textured objects using the BlobTree model where initial and target objects are represented in 4D space. Geometry and texture transformations are applied simultaneously using a warping function for geometry and one of their blending functions for textures. Another method for metamorphosis between textured objects using the unstructured lumigraph representation by Ludwig et al. [LBTM15], achieved texture transformation using a simple linear interpolation, influencing output quality.

The main contributions of this work are as follows:

- The introduction of the heterogeneous STB concept in which the transformation of geometry and attributes happens simultaneously and interconnectedly in the higher dimensional STB space using STTI for attribute interpolation;
- The extension of the HFRep introduced in [TAFP19] to take advantage of benefits provided by the unification of different representations such as function representation (FRep), signed distance fields (SDFs) and adaptively sampled distance fields (ADFs) [FPRJ00] while compensating for their drawbacks;
- A step-by-step algorithm that realises STB for heterogeneous objects in the interconnected set of higher-dimensional geometric and colour spaces with subsequent projection to the initial geometric and colour spaces;

As proof-of-concept we show how the developed method can be applied to textured cubist shapes [CMPA18].

2. Space-time blending with attributes

We introduce the concept of heterogeneous STB where geometry and attribute transformations occur simultaneously in a higher specific STB dimension. In this work we only consider colour transformations as attributes. For defining heterogeneous objects we use SDFs and HFRep [TAFP19] extended with ADFs.

2.1. Hybrid function representation

Tereshin et al. [TAFP19] introduced the concept of HFRep for heterogeneous objects, based on unifying advantages of FRep and SDFs while simultaneously compensating for their drawbacks. Here we extend the HFRep representation by unifying FRep with SDFs and ADFs. The ADFs are generated using a hierarchical subdivision of Euclidean space according to the geometry defined by the FRep function. The distance values are obtained in corner points of the generated cells and the resulting field is restored using continuous spline-based interpolation, e.g. PHT-splines.

To define object with its attributes we need to use the concept of hypervolumes [PASS01]. A hypervolume object is defined as a tuple $O = (G, A_1, ..., A_n)$, where G is a point set in n-dimensional Euclidean space E^n and A_n is an object attribute. The HFRep object can be obtained using equation:

$$F_{HFRep}(\mathbf{x}) = S(F_{FRep}(\mathbf{x}))F_{UDF}(\mathbf{p}, \mathbf{p_s}) \quad (1)$$

where $S(F_{FRep}(\mathbf{x}))$ is a step-function following the FRep function $F_{FRep}(\mathbf{x})$ behaviour, providing the sign to the unsigned distance function (UDF) $F_{UDF}(\mathbf{p}, \mathbf{p_s}) = \inf_{\mathbf{p_s} \in S} d(\mathbf{p}, \mathbf{p_s})$, where $\mathbf{p}$ is a given point in Euclidean space and $\mathbf{p_s}$ is a point belonging to the surface S of the object, with $\mathbf{x} = (x, y, z)$ being a point in Euclidean 3D space.

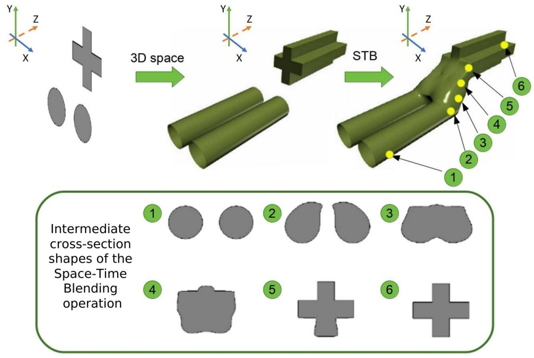

Figure 1: *The concept of space-time blending: two given 2D shapes (two disks and a cross, top left) are extended to 3D space as half-cylinders (top centre) with a gap between them. Then a blending union operation is applied, adding material (top right).*

2.2. Space-time blending

The metamorphosis between initial and target objects O_1 and O_2 in this work is based on the well-established STB method [PPK04], a geometric operation (Fig. 1) of bounded blending performed in a higher dimensional space, where the last coordinate is associated with time t. The gradual shape transformation happens in the time interval $t \in [0, 1]$, where, at $t \leq 0$, only O_1 exists, then at $t > 0$ it disappears, and at time $t = 1$, O_2 appears and exists for any $t \geq 1$.

Let initial and target objects O_1 and O_2 be defined in a higher dimensional space by distance based functions $f_1(\mathbf{x}, t)$ and $f_2(\mathbf{x}, t)$ respectively. Then $\mathbf{x} = (x, y, z)$ is a point in Euclidean 3D space and the resulting STB function $F_b(f_1, f_2, f_3)$ between the two objects is

$$F_b(f_1, f_2, f_3) = F(f_1, f_2) + a_0 disp_b(d_r(f_1, f_2, f_3, a_1, a_2, a_3)) \quad (2)$$

where $disp_b(d_r(f_{1_t}, f_{2_t}, f_{3_t}))$ is a displacement function, $d_r(f_1, f_2, f_3)$ is a generalised distance function and $F(f_1, f_2)$ is a set-theoretical union of two shapes defined by the R-functions [PASS01]. The resulting blending shape is affected by the bounding solid defined by the function $f_3(\mathbf{x}, t)$. Coefficients $a_0, a_1, a_2, a_3 \in \Re$ are non-zero numerical parameters.

2.3. Object attribute interpolation

In this work, we consider coloured or textured heterogeneous objects that are defined using SDFs and HFReps, using a voxel data structure to handle both geometry and attributes. To enable transformation of colours in a higher dimension to happen simultaneously with the geometry transformation relying on STB, we use STTI [SFA*15], as this works in the higher dimension and can also deal with complex textures. The basic STTI can be defined as:

$$\mathbf{c}(\mathbf{x}, t) = \omega_1(\mathbf{x}, t)\mathbf{c_1} + \omega_2(\mathbf{x}, t)\mathbf{c_2} \quad (3)$$

$$\omega_1(\mathbf{x}, t) = \frac{f_2(\mathbf{x}, t)}{f_1(\mathbf{x}, t) + f_2(\mathbf{x}, t)} \quad \omega_2(\mathbf{x}, t) = \frac{f_1(\mathbf{x}, t)}{f_1(\mathbf{x}, t) + f_2(\mathbf{x}, t)}$$

$$\mathbf{c_i}(\mathbf{x}) = \frac{\sum_{j=1}^{N} \tilde{w_j}(\mathbf{x})\tilde{\mathbf{c_j}}}{\sum_{j=1}^{N} \tilde{w_j}(\mathbf{x})}; \quad \tilde{w_j}(\mathbf{x}) = \frac{1}{\widehat{f_j}(\mathbf{x})}, \quad i = 1, 2 \quad (4)$$

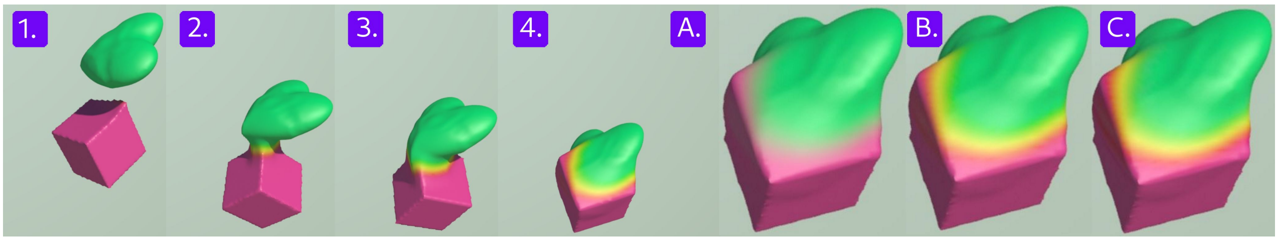

Figure 2: *Result of applying heterogeneous STB to an SDF cube object and an HFRep heart object. The result of applying (A) STTI for RGB colour interpolation; (B) STTI for HSV colour interpolation; (C) linear HSV colour interpolation. Also see the supplementary videos.*

where $\mathbf{c_1}$ is the colour of the input object O_1, $\mathbf{c_2}$ is the colour of the target object O_2, $t \in [0,1]$ and $\omega_1(\mathbf{x},t)$ as well as $\omega_2(\mathbf{x},t)$ are weights. Expression (4) is used when the object has multiple coloured partitions, and $\widehat{f_j}(\mathbf{x})$ is a function defining the quadratic Euclidean distance from the point $(\mathbf{x})$ to the voxel centre.

For colour interpolation we use the HSV colour space as it provides visually appealing results, and RGB for comparison. The main advantage of HSV is that the colours are represented in a continuous manner, controlled by the hue value $H \in [0,360^o]$.

2.4. Basic algorithm

We consider colour as an attribute defined for initial and target objects O_1 and O_2, assuming that the colour and geometry spaces are interconnected and their transformation happens simultaneously in a higher dimensional STB space. Our basic algorithm approach is:

1. The definition of initial and target heterogeneous objects (Fig. 2, frame 1) as hypervolumes using some distance-based functions $d_1(\mathbf{x},A_1)$ and $d_2(\mathbf{x},A_2)$ with attribute sets A_1 and A_2. We split these into two parts: geometric $d(\mathbf{x})$ and attribute $A(a_1,a_2,...,a_n)$, where $\mathbf{x}$ is a point in n-dimensional Euclidean space. The geometric part is defined using equation (1) or as

$$d(\mathbf{p},\mathbf{p_s}) = sign(\mathbf{p}) \cdot \inf_{\mathbf{p_s}\in S} d(\mathbf{p},\mathbf{p_s}) \tag{5}$$

where $d(\mathbf{p},\mathbf{p_s})$ is the minimal signed Euclidean distance between two points, $\mathbf{p}$ is any given point and $\mathbf{p_s}$ is a point belonging to the surface S of the object. Both objects can be a combination of different DF types, e.g. SDFs or HFReps.
2. The use of attribute functions (e.g. procedural) A_1 and A_2 to define attribute (colour) distribution according to the defining functions. We consider that attribute functions A_1 and A_2 for both objects O_1 and O_2 are defined as colour fields $A_1(a_1,...,a_n) = c_1(r_1,g_1,b_1)$ and $A_2(a_1,...,a_m) = c_2(r_2,g_2,b_2)$. These are defined for each point of the objects O_1 and O_2 with the defining functions $d_1(\mathbf{x}) \geq 0$ and $d_2(\mathbf{x}) \geq 0$.
3. To apply STB to both objects O_1 and O_2, we raise their defining functions to the higher dimension and make them dependent on time t, applying an additional STB operation (2) to each object to obtain a smooth transition between them:

$$\begin{aligned}
&O_1: \ f_1(\mathbf{x},t) = f_{DF_1}(\mathbf{x},t); \quad f_2(\mathbf{x},t) = -t;\\
&f_3(\mathbf{x},t) = t + P_c, \quad t \geq -P_c;\\
&O_2: \ f_1(\mathbf{x},t) = f_{DF_2}(\mathbf{x},t); \quad f_2(\mathbf{x},t) = t-1;\\
&f_3(\mathbf{x},t) = P_c - t, \quad t \leq P_c;\\
&F_b(f_1,f_2,f_3) = f_1 \wedge f_2 + a_0 disp_b(f_1,f_2,f_3,a_1,a_2,a_3)
\end{aligned} \tag{6}$$

where $f_1(\mathbf{x},t)$ is the DF function raised to a higher-dimension, defining either object O_1 or object O_2, $f_2(\mathbf{x},t)$ is the function defining a smoothing object as the subtraction of the negative half-space along time-axis t, $f_3(\mathbf{x},t)$ is the bounding function restricting the area of the STB bounding intersection ($\wedge$) and P_c is the position of the cutting plane on time-axis t.
4. For each point in which blending function $F_b(f_1,f_2,f_3) \geq 0$ and $f_1 < 0, f_2 < 0$, any attribute operation transformation can be applied in the general case. Here we apply STTI (3-4), which is also executed in a higher dimension. Colours c_1 and c_2 can be defined in RGB or HSV colour spaces. If HSV colour space is used, we need to find the shortest path between two 'hue' values and linearly interpolate between these, weighted by w_2, (3) while using the STTI approach for 'saturation' and 'value' values. Finally, STTI is applied and the resulting colour c is converted back to RGB (see Fig. 2, B).

We compare colour interpolation using RGB and HSV in Fig. 2, A-C. Fig. 2A shows the result of applying STTI in RGB colour space, 2B depicts HSV colour space and 2C demonstrates how linear interpolation works in HSV colour space. As expected, STTI in HSV provides smoother colour transitions than linear interpolation.

3. Implementation and results

In this work, we broaden the concept of 4D Cubism [CMPA18], an art application providing tools for creating artistic shapes in a cubist style employing the STB method that could only deal with geometric metamorphosis without attributes, by introducing attribute transformations and the concept of heterogeneous STB for metamorphosis between volumetric objects. We also apply this concept to the objects defined by different types of DF based representations, e.g. SDFs and HFReps. All examples are implemented in SideFX Houdini using the OpenVDB library for handling both attributes and geometry transformations, rendered using an Intel Xeon E5-1650 3.20 GHz PC with 32 Gb of RAM.

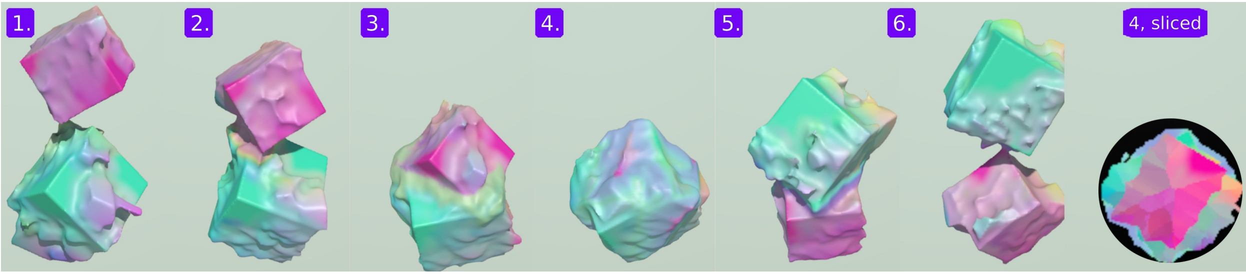

Figure 3: *Frame sequence of heterogeneous STB applied to two heterogeneous objects with multiple features defined by HFRep and SDF.*

3.1. Dynamic 4D heterogeneous cubism

We demonstrate our approach using two examples. The first example (Fig. 2) applies heterogeneous STB to a heterogeneous oscillating SDF cube object and an HFRep heart object.

In the basic algorithm, we first define the initial (SDF cube) and target (HFRep heart) objects. We then define the colour attributes for both objects. Next we raise both defining functions for the cube and the heart objects to a higher STB dimension using equations (6). Finally, we apply the introduced heterogeneous STB to both objects and compute both in-between geometry and colour attribute transformations simultaneously using RGB (see Fig. 2, a) and HSV (see Fig. 2, b) colour spaces for attribute transformations. We implemented interpolation in HSV using linear interpolation and STTI interpolation. As can be seen in Fig. 2, STTI provides a smoother colour transition than linear interpolation and using HSV results in smoother transition between colours than RGB.

In Fig. 3 we show a more complex example of two oscillating heterogeneous objects with cubist coloured features demonstrating that HFRep objects can easily be combined with SDF objects. First, two polygonal cubes are converted to OpenVDB objects. Then SDF functions for both cubes are obtained. To specify where coloured features will be added both cubes are subdivided using an octree. Some of the features are defined using HFRep , some using SDF. Next we specify colour attributes for the cubes and defined features. At the third step, we combine functions that define features with functions specifying the base cubes, using set-theoretical operations, and raise the resulting functions to a 4D space using equations (6). Finally we compute simultaneously STB and STTI (HSV colour space) and then map the STTI result computed in a separate voxel grid to the obtained geometry. Note that operations on SDFs and HFReps cannot preserve attributes, so these are conducted in separate vectorised OpenVDB voxel grids.

4. Conclusions

We have presented the concept of heterogeneous space-time blending based on STB for handling geometry transformation and using STTI for handling attribute transformations. In the general case, instead of STTI, any method that is suitable for attribute transformation can be used. We have also presented the basic algorithm for implementing heterogeneous STB that we further applied to dynamic objects (4D Cubism [CMPA18]). For representing heterogeneous objects we used the distance based representations SDFs and HFRep, briefly introducing the concept of HFRep that is based on FRep and ADFs. As future work we are planning to develop new methods for defining attributes and their transformations. We would also like to broaden the class of attributes that we can deal with using our approach, e.g. volumetric materials and transparency.

References

[BGA05] Barbier A., Galin E., Akkouche S.: A framework for modeling, animating, and morphing textured implicit models. *Graph. Models 67*, 3 (2005), 166–188. 1

[BW01] Breen D. E., Whitaker R. T.: A level-set approach for the metamorphosis of solid models. *IEEE T VIS COMPUT GR. 7*, 2 (2001), 173–192. 1

[CMPA18] Corker-Marin Q., Pasko A., Adzhiev V.: 4d cubism: Modeling, animation, and fabrication of artistic shapes. *IEEE Computer Graphics and Applications 38*, 03 (2018), 131–139. 2, 3, 4

[DYTT05] Dinh H. Q., Yezzi A., Turk G., Turk G.: Texture transfer during shape transformation. *ACM TOG 24*, 2 (2005), 289–310. 1

[FPRJ00] Frisken S. F., Perry R. N., Rockwood A. P., Jones T. R.: Adaptively sampled distance fields: A general representation of shape for computer graphics. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques* (2000), SIGGRAPH '00, pp. 249–254. 2

[LBTM15] Ludwig M., Berrier S., Tetzlaff M., Meyer G.: 3d shape and texture morphing using 2d projection and reconstruction. *Computers & Graphics 51* (2015), 146–156. 2

[PASS01] Pasko A., Adzhiev V., Schmitt B., Schlick C.: Constructive hypervolume modeling. *Graph. Models 63*, 6 (2001), 413–442. 2

[PPK04] Pasko G., Pasko A., Kunii T.: Space–time blending. *Computer Animation and Virtual Worlds 15*, 2 (2004), 109–121. 1, 2

[SdGP*15] Solomon J., de Goes F., Peyré G., Cuturi M., Butscher A., Nguyen A., Du T., Guibas L.: Convolutional wasserstein distances: Efficient optimal transportation on geometric domains. *ACM TOG 34*, 4 (2015), 66:1–66:11. 1

[SFA*15] Sanchez M., Fryazinov O., Adzhiev V., Comninos P., Pasko A.: Space-time transfinite interpolation of volumetric material properties. *IEEE T VIS COMPUT GR. 21*, 2 (2015), 278–288. 1, 2

[TAF*20] Tereshin A., Adzhiev V., Fryazinov O., Marrington-Reeve F., Pasko A.: Automatically controlled morphing of 2d shapes with textures. *SIAM Journal on Imaging Sciences 13*, 1 (2020), 78–107. 1

[TAFP19] Tereshin A., Adzhiev V., Fryazinov O., Pasko A.: Hybrid Function Representation with Distance Properties. In *Eurographics 2019 - Short Papers* (2019), pp. 17–20. 1, 2

[WCX*13] Weng Y., Chai M., Xu W., Tong Y., Zhou K.: As-rigid-asî" possible distance field metamorphosis. *Computer Graphics Forum 32*, 7 (2013), 381–389. 1