

# Creation of paper craft model from 3D entity source

Ivan Škultéty  
Comenius University, Slovakia  
Faculty of Mathematics, Physics and Informatics,  
KPGSO  
[www.cray.sk/skultety/diplom](http://www.cray.sk/skultety/diplom)  
[skultety@cray.sk](mailto:skultety@cray.sk)

Mgr. Martin Samuelčík  
Comenius University, Slovakia  
Faculty of Mathematics, Physics and Informatics,  
KPGSO  
[www.sccg.sk/~samuelcik](http://www.sccg.sk/~samuelcik)  
[samuelcik@fmph.uniba.sk](mailto:samuelcik@fmph.uniba.sk)

## Abstract

The aim of our work is to create an efficient algorithm useful for conversion 3D entity source like 3ds object to planar representation and after that we will use it for creating a hard-copy (paper craft) model of object. The most important milestone of my work is to explode the boundary representation and optimal layout of result polygons.

**Keywords:** boundary representation, optimal layout, 3DS files, paper-craft

## 1 Introduction

As an input we have 3D model (later in our work we will focus on buildings like castle, historical building or cottage). In this paper we will use game cube for demonstration. The first step is to pre-process source to reduce small parts, because small parts are hard to model from paper. These parts are not deleted entirely, but will be used in texture. After this step we have the skeleton of the model so we are ready to transform it into boundary representation. This is first really important step because since that we have suppressed all advantages or disadvantages of input format and we have pure data of model with additional information of appearance (texture or colour information mapped for define faces) and these data are independent on input format.

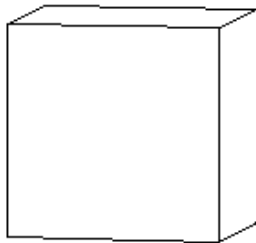


Figure 1: 'Clear' model

Next milestone may be achieved when we explode boundary model to set of polygons, but we have to remember relation between faces.

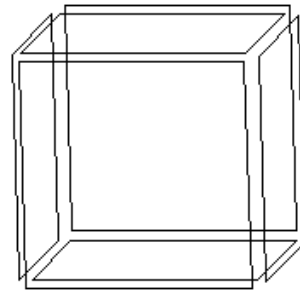


Figure 2: Explode of B-rep. model

After that we have set of polygons with their relation information, relative or absolute mapping of textures or any other appearance information. We would like to find the best layout for the polygons on the paper (printed paper(s) will be our result). For this purpose Capacity algorithm was chosen. One by one polygons are connected to the related polygons and placed into paper. While performing this placing, we have to check for following conditions and optimizations:

- Placed faces cannot intersect, only at the edges of faces.
- For connecting of faces that are not connected in placement and are connected in 3D model we have to add small connecting field on one face. These connecting parts can be on both faces.
- Connection of faces in placement is made only if relation of faces from 3D model exists.
- We can have more than one component of continuity.
- While placing faces we want to check for usage of paper, e.g. we want to minimize empty space around faces.
- Placing algorithm should start with largest faces.
- Optionally we can try to place similar faces near each other.

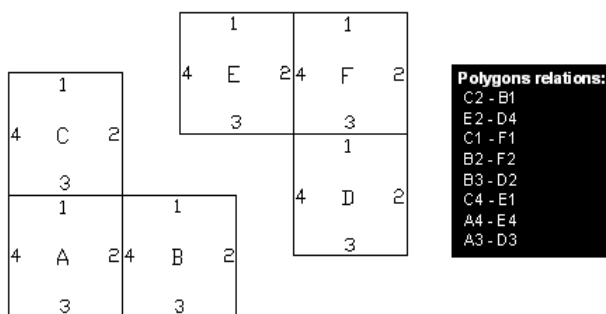


Figure 3: The best layout for cube with relation information

The last milestone of our work is to create map of the model with description how to connect polygons or how to turn their parts according defined angels to achieve the best result.

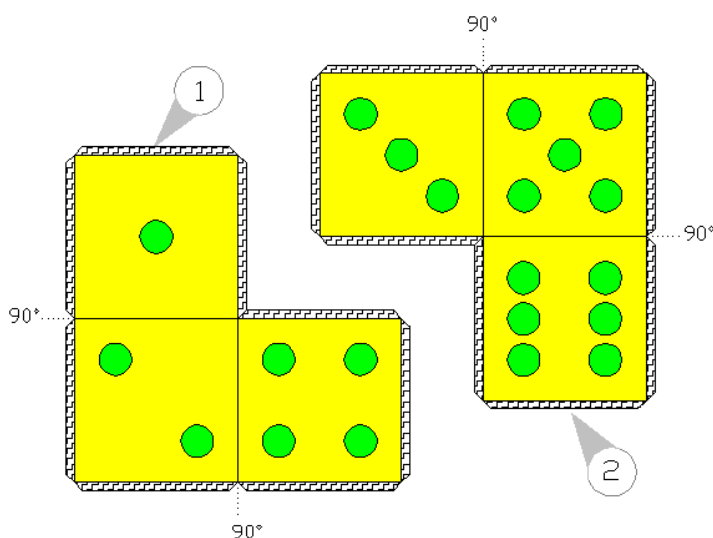


Figure 4: Final polygons layout with descriptions.

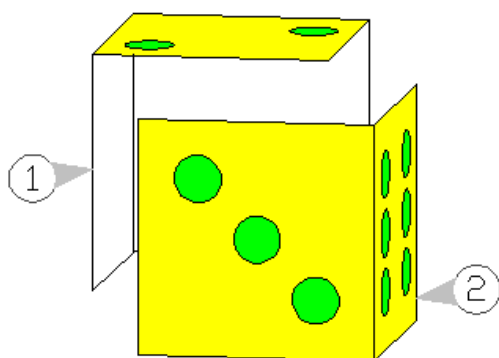


Figure 5: Model map.

## 2 Implementation

Application for converting 3D model is already done with basic functionality. Input source model could be 3ds file (models created in 3D Studio Max®). After loading source the graphics interface via OpenGL® libraries

display the model and user can manipulate it with rotating, scaling, applying alfa-blending, or turning on/off textures. OnClick event on tab called 'Result' invoke the method for transformation and after few seconds or minutes the result is displayed.

## Conclusion & future work

The game cube example has illustrative purpose (it is the simplest solid model) and show us how complicated can be transformation process from 3 dimensional models into planar. The whole problem is even more complicated for spherical or bulbous shape models. Triangulation or Platonic solids representation are not good ideas because of too difficult workflow without good results. Another possible way is to cut the atypical model into many stripes like an orange. This seems to be the best of all possible solutions (at least it is often used). Disadvantage is hard way of texture mapping into the result.

The best way is to implement all possibilities and put the decision on the user.

In the future we also plan to support wide range of input file formats e.g. dwf, vrml etc.

## References

- VYŠKOVSKÝ RICHARD: Městská památková rezervace, ABC mladých techniků a přírodovedců, Numbers 4,5, Volume 36, Mladá Fronta
- LEWIS JEFF: 3D Studio File Format, <http://www.the-labs.com/Blender/3dsspec.html>
- ŽÁRA JIŘÍ, BENEŠ BEDŘICH and FELKEL PETR: Moderní počítačová grafika, Computer Press, 1998
- ANDREA L. AMES, DAVID R. NADEAU, JOHN L. MORELAND: VRML 2.0 Sourcebook, John Wiley & Sons, Inc., 1997
- NeHe PRODUCTIONS: OpenGL tutorial, <http://nehe.gamedev.net>