

Digital Relief Generation from 3D Models

Meili Wang¹, Yu Sun¹, Min Jiang², Kun Qian², Jian Chang², and Dongjian He^{3*}

1 College of Information Engineering, Northwest A&F University, Yangling, Shaanxi, 712100, China

2 National Center for Computer Animation, The Media School, Bournemouth University, UK

3 College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling, Shaanxi, 712100, China

Abstract: Unlike generating reliefs from images and range data, this paper proposes an efficient method to generate both bas-reliefs and high-reliefs directly from three-dimensional (3D) meshes. To produce relief features that are visually appropriate, the 3D meshes must be scaled. In our proposed method, 3D unsharp masking is used to enhance the visual features in the 3D mesh. We compare average smoothing and Laplacian smoothing to achieve better smoothing results, and then employ a nonlinear variable scaling scheme to generate the final bas-reliefs and high-reliefs. Using our method, relief models can be generated from arbitrary viewing positions with different gestures and combinations of multiple 3D models. The generated relief models demonstrate that our method is efficient and effective.

Key words: high-relief; bas-relief; mesh enhancement; scaling.

1 Introduction

Reliefs are sculptured artworks where a modeled shape is raised or lowered and usually attached to a planar background^[1]. There are three main types of reliefs: high-reliefs, bas-reliefs and sunken-reliefs. In a high-relief, the raised sculpture is over 50% of the scale height of what is being represented. Bas-reliefs, by comparison, are much more compressed, and are suitable for scenes with many figures, landscapes or architectural backgrounds; e.g., on most coins, Sunken-reliefs, the figures are actually carved into the surfaces.

Existing relief models created by artists are a complex procedure. Once the relief modes have been produced it is not easy to modify or maintain them. Such reliefs also only represent a single viewing point. Animators and artists, though, want a digital relief-an object having the characteristics of a real relief but with the advantages of being virtual. There are already many 3D models in the public domain that could be processed into reliefs and be easily amended and refined. A system to produce digital reliefs

must allow for interactive design and generate standard 3D relief representations.

Previous relief generation research has been mainly concerned with bas-relief generation as it is very challenging. The general approach is to start with height field data or a 3D model, and then apply image processing techniques to produce the relief^[2]. In this paper we propose a method to simultaneously generate digital high-reliefs and bas-reliefs. We develop a boosting algorithm with different smoothing schemes to enhance the geometrical details, and employ a nonlinear method to scale the 3D meshes while preserving their features.

2 Related work

Reliefs can be generated from direct modeling, images and 3D models. Direct modeling need special expertise and is a labor-intensive process.

Bas-relief generation from images: Alexa and Matusik^[3] proposed a method to generate bas-reliefs from a pair of input images using an iterative least square optimization to reconstruct the relief surfaces.

Wang^[4] presented an algorithm for bas-relief generation from a 2D image with gradient operations, such as magnitude attenuation and image enhancement, which greatly improved the quality of the relief. Ji^[5] presented a novel

* This work is partially supported by National Natural Science Foundation (61402374), the National High-tech Research and Development Program (863 Program: 2013AA10230402), and China Postdoctoral Science Foundation.

method for digital bas-reliefs modeling from a composite normal image. In their work, several layers of normal images were composed into a single normal image. Li^[6] also presented an image-based relief generation method, which was designed especially for brick and stone relief. They provided a two-level approach for height map estimation from single images, with a PDE-based mesh deformation scheme and normal computation. However, it is difficult to extend image-based relief generation to high-relief generation as the images contain insufficient height information and the computation time is much greater.

Model-based relief generation: Reliefs generated from 3D objects have been considered a promising way to create bas-reliefs, and also allow for the reuse of existing 3D models. The final generated reliefs convey real height information, which could be easily machined directly into real reliefs. Furthermore, the generated 3D reliefs can be edited and modified before real machining. The challenge is to retain the fine details of a 3D object while greatly compressing its depths to produce an almost planar result. To achieve the highest quality results, sharpness, richness of detail and accuracy should be taken into account. The existing techniques use different feature preserving methods to generate bas-relief models with rich details, even with a high compression ratio^[7].

Zhang^[8] proposed a series of gradient-based algorithms, including height field deformation, high slope optimization and fine detail preservation, which could be used to generate bas-reliefs interactively. Recently, Schüller^[9] proposed a unified framework to create bas-reliefs from target shapes, viewpoints and space restrictions.

Most of the methods mentioned above first convert 3D models into height fields, which can only generate bas-reliefs. Arpa^[10] presented a method for synthesizing high reliefs using differential coordinates. High-relief synthesis is semi-automatic and can be controlled by user-defined parameters to adjust the depth range and the placement of the scene elements with respect to the relief plane. Our proposed method can generate both high-reliefs and bas-reliefs from 3D models with high efficiency, making it computationally feasible to obtain different viewpoints under different gestures of a

single model and also to combine models together to generate more multiple relief models.

3 Mesh enhancement and reliefs generation

When a 3D mesh is provided, the algorithm first applies mesh enhancement by 3D unsharp masking (USM) to flatten the mesh while preserving its details.

3D USM is implemented first by a simple smoothing scheme. The smoothed mesh is subtracted from the original mesh to extract the mesh's higher order features. These features can be enhanced by scaling them and adding them back to the original mesh^[11]. After enhancing the features, a proportional nonlinear scaling scheme is adopted to produce the final bas-reliefs and high-reliefs with different scaling factors.

3.1 Unsharp Masking

USM is a feature enhancement technique in image processing, which splits a signal into low frequency and high frequency components. An input image is convolved with a low pass kernel resulting in a smooth version of the input image. Subtracting the smooth version from the original image leads to a high frequency image containing peaks at a small-scale level of detail. Adding a multiple of the high frequency image back to the smooth image emphasizes the fine structures in the newly reassembled image.

3D USM is an extension and derivative of 2D USM. Its main function is to extract the mesh's higher order features. In the 3D case,

$$M_{sharp} = M_o + \delta * (M_o - M_{smooth}) \quad (1)$$

where M_{sharp} is the sharpened mesh, M_o is the base mesh, M_{smooth} is the smoothing mesh with different smoothing strategies, δ is the amount of enhancement. The adjustment of δ could be useful for artistic control.

3.2 Mesh smoothing strategies

(1) Average smoothing

The simplest approach is average smoothing, which is widely used in image processing. It replaces each vertex with the average of its one-ring vertices. It is straightforward to implement and has high computational efficiency, but can cause blurring. It is expressed as in Eq.2

$$P_{in} = P_{io} + \sum_{Q \in N(P_{io})} Q \quad (2)$$

where P_{in} is updated vertex of P_i , P_{io} is original vertex of P_i , and $N(P_{io})$ are the one-ring neighbors of P_{io} .

(2) Laplacian smoothing

Laplacian smoothing changes the position of nodes without modifying the topology of the mesh. It is simple to implement because a given vertex only requires information about its immediate neighbors. For each vertex in a mesh, a new position is chosen based on local information and the vertex is gradually moved there. There are a number of different approximations for the Laplacian operator, and each one has a particular use. Desbrun (1999) describes its discrete approximation as

$$L(x_i) = \frac{1}{\sum w_{ij}} \sum_{j \in N(i)} w_{ij}(x_j - x_i) \quad (3)$$

where w_{ij} is the weight for the edge with end points (x_i, x_j) . There are several schemes for defining the weights w_{ij} : combinatorial, distance, and conformal weights. The combinatorial weight is an adjacency matrix, and any arbitrary edge weight is the Euclidean distance of the two faces connected by the edges.

Distance weight is the nonlinear operator

$$w(p_i) = \frac{1}{E} \sum_{q \in N(i)} \frac{q - p_i}{|q - p_i|}, \quad E = \sum_{j \in N(i)} |q - p_i| \quad (4)$$

where $|q - p_i|$ is the distance between vertex q and p_i , $w(p_i)$ is the weight matrix between p_i and its one-ring neighbors and E is the distance sum of p_i and its neighbors.

In the conformal weight approach, for a mesh with triangular faces, if three vertices are v_i, v_j, v_k , then the angle between edge (v_k, v_i) and edge (v_k, v_j) can be expressed as angle $(v_k - v_i, v_k - v_j)$, so weight (i, j) can be calculated as

$$w(i, j) = \cot(\text{angle}(v_k - v_i, v_k - v_j)) \quad (5)$$

In Eq.5, vertex v_k and v_j belong to one-ring neighbors of vertex v_i . This weight scheme smoothes while preserving shape, minimizing potential distortion related to angle changes. Laplacian smoothing can be applied iteratively

to get the desired mesh smoothness. So in the following experiments, the conformal weight approach has been applied.

3.3 Nonlinear scaling scheme

To scale 3D meshes efficiently, a nonlinear scaling should be used instead of linear scaling because linear scaling cannot distinguish the important part of the mesh. The objective is to scale the height of the relief proportional to height field. To generate a relief, the higher part of the object should be scaled more than the lower part to preserve the features. An attenuation function can be used for this purpose, which is expressed by the following equation^[13].

$$G(h) = SF * \left(\text{sgn}(h) \frac{\alpha}{|h|} \left(\frac{|h|}{\alpha} \right)^\beta \right) \quad (6)$$

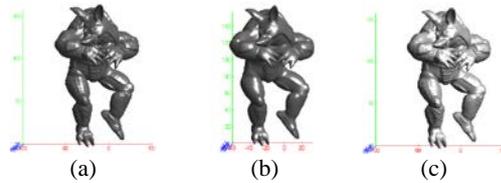
The scaling factor SF controls the depth compression ratio, where a large value means less compression. For bas-relief generation, a small value is chosen to present the model in a nearly flat plane and a relatively large value applies for high-relief. SF also defines the level of preservation of details, where a small value would smooth out small depth changes in the model. The parameter α determines to what degree height has changed. Larger heights are scaled more (assuming $0 < \beta < 1$). Choosing $\alpha = 0.1$ and $\beta = 0.9$ gives the scaling suggested by^[13].

4 Results of Relief Generation

4.1 High reliefs

In high-reliefs, the height have over half of their natural depth attaching to a surface or separating from it. Therefore, SF is defined to be >0.5 according to the definition.

Armadillo is taken as an example to illustrate the results of the algorithm. For high-relief generation, the parameters are: $\delta = 0.2$, $SF = 0.7$, $SF = 0.6$, $SF = 0.5$. All testing models are from Aim Shape (2009).



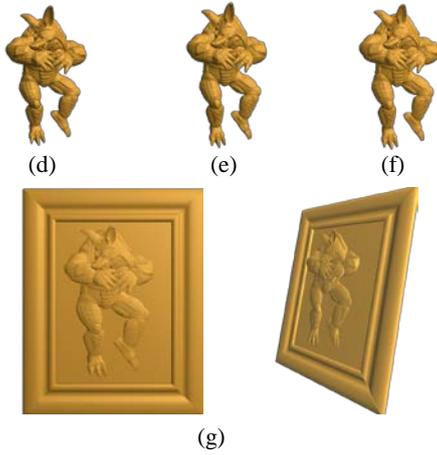


Figure 1. High-reliefs with Armadillo model. (a), (b), (c) represent the input, smoothed and sharpened models respectively. (d), (e), (f) are different final relief result with different parameters. (g) is the rendered and rotated model of (f).

It can be seen from Figure 1 that a higher SF (0.7) preserves more features and the overall details are preserved for all SF . The influence of δ will be examined in Section 4.4.

4.2 Bas-relief

In bas-reliefs, the height have under half of their natural depth. Recent studies have assumed a compression ratio of 0.02. Figure 2 demonstrates the bas-reliefs of the Armadillo models generated by the proposed method with the parameters $\delta = 0.2$, $SF = 0.02$. The mesh enhancement techniques and non-linear scaling scheme effectively preserve the details of the models in bas-relief.



Figure 2 Armadillo bas-relief generated by the proposed algorithm.

Figure 3 are examples showing the performance of the proposed algorithm on Raptor and Ball. It can be seen that it is able to generate bas-relief with details preserved.

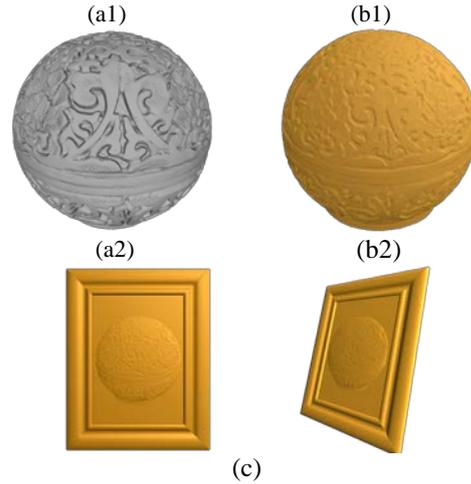


Figure 3. Bas-reliefs with different models. (a1), (a2) represent the input models, (b1), (b2) are bas-reliefs accordingly. (c) is the rendered and rotated model of (b2).

4.3 Multiple views and multiple models

It is straightforward to apply different viewing angles under different gestures to the original model because the proposed method operates on 3D meshes directly. For example, adjusting the original model to get different viewpoints under different gestures of the model and then applying the proposed algorithm generates bas-reliefs with different view angles, taken Armadillo as the example (see Figure 4).

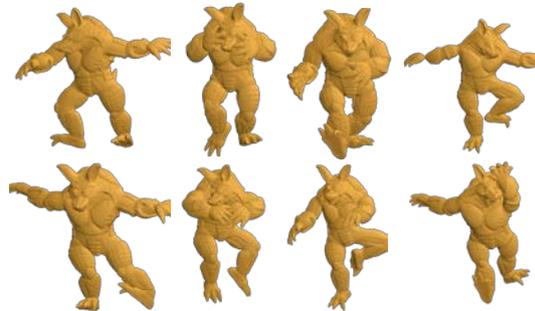
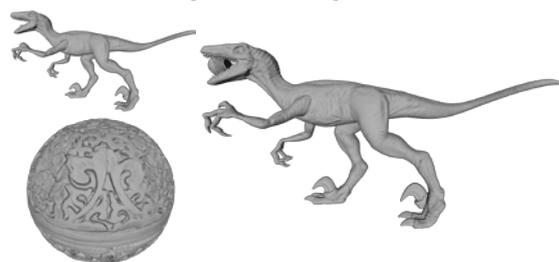


Figure 4. A serials of bas-reliefs with different gestures and viewpoints of a single model.

Generating complex combined relief models is also an intended extension of the proposed method. Putting different models into a meaningful scene to generate multiple models is illustrated in Figure 5 and Figure 6.



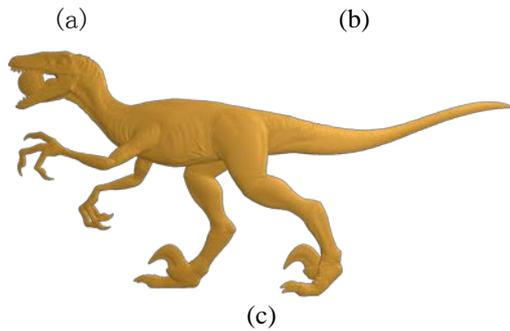


Figure 5. Bas-reliefs with multiple models. (a) represents two single models of Raptor and Ball, (b) represents the combined model of Raptor and Ball and (c) is the final relief results.

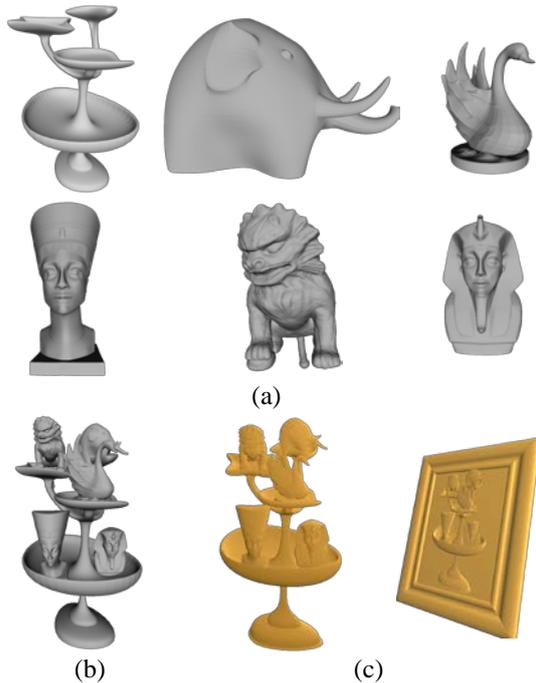


Figure 6. Bas-reliefs with multiple models. (a) represents six pieces single models, (b) represents the combined model and (c) is the final relief result and the rotated one.

4.4 Smoothing Comparison and Parameter Testing

Average smoothing is easy to implement but the smoothing result cannot be changed and sometimes cause blurring (Figure 7(b)). While Laplacian smoothing is relatively easy to implement compared to bilateral filtering and cross-bilateral filtering, but the smoothing results can be adjusted iteratively (Figure 7(c)), accordingly influence the final relief models.

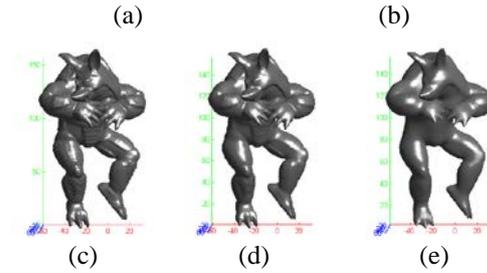
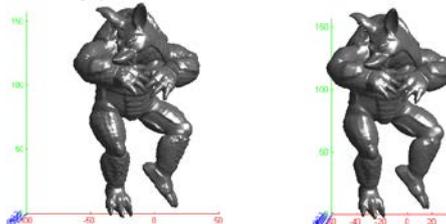


Figure 7 Two Smoothing strategies results. (a) is the original model, (b) is after average smoothing (c)-(e) are after Laplacian smoothing with different parameters.

We now find the parameter that most strongly influences the results and determine its appropriate value. The test case will be on bas-relief generation of the Skull model. We vary the amount of enhancement δ :

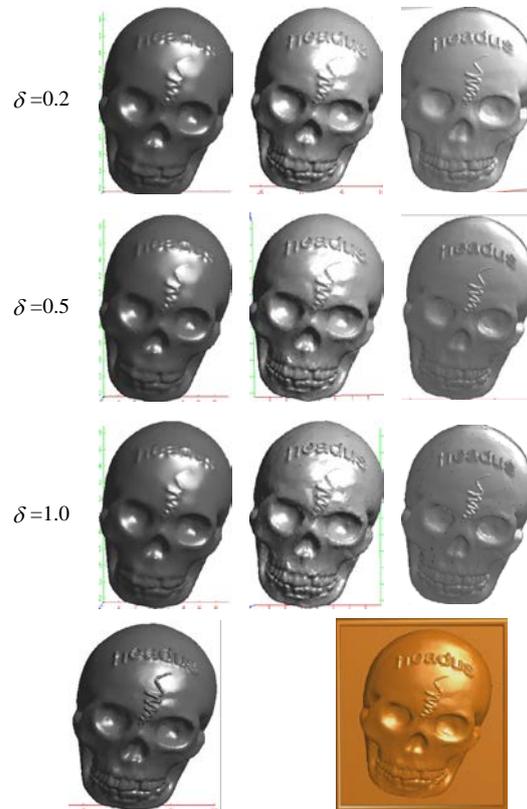


Figure 8. Different δ from 0.2 to 1.0.

It can be seen from Figure 8 that with the increase of δ , the relief models are enhanced. However, the larger δ results in undesired deformation when applied to 3D USM. In a generic USM, the high frequency part is linearly scaled by a specified factor and added back to the original one. A suitable specified factor, in this case δ , should be chosen carefully because artifacts are triggered by a larger δ in the final bas-relief generation, as seen in Figure 8.

Esthetically pleasing results are produced with $\delta = 0.2$.

5 Conclusion

The method proposed in this paper can produce reliefs directly from 3D meshes. 3D USM and the nonlinear scaling scheme have been developed to generate bas-reliefs and high-reliefs. The results have shown that this method can preserve the original details in both relief types. Compared with previous approaches, the proposed method is relatively easy to implement because it operates on 3D meshes directly. Therefore, our method is amenable to relief generation from multiple viewpoints under different gestures of the original model and also can be extended to relief generation of multiple models.

References:

- [1] Flaxman J, Westmacott R. Lectures on Sculpture..[M]. Henry G. Bohn, 1838.
- [2] Kerber J, Tevs A, Belyaev A, et al. Feature sensitive bas relief generation[C]//Shape Modeling and Applications, 2009. SMI 2009. IEEE International Conference on. IEEE, 2009: 148-154.
- [3] Alexa M, Matusik W. Reliefs as images[J]. ACM Trans. Graph., 2010, 29(4): 60:1-60:7..
- [4] Wang, M., Chang, J., Pan, et al., 2010. Image-based bas-relief generation with gradient operation. In: The Eleventh IASTED International Conference on Computer Graphics and Imaging, 17-19 February 2010, Innsbruck, Austria.
- [5] Ji Z P, Ma W Y, Sun X F, Bas-relief modeling from normal images with intuitive styles, IEEE Transactions on Visualization and Computer Graphics, Vol. 20, No. 5, May 2014, pp. 675-685.
- [6] Li Z, Wang S, Yu J, et al. Restoration of brick and stone relief from single rubbing images[J]. Visualization and Computer Graphics, IEEE Transactions on, 2012, 18(2): 177-187.
- [7] Kerber J, Wang M, Chang J, et al. Computer assisted relief generation—A survey[C]//Computer Graphics Forum. Blackwell Publishing Ltd, 2012, 31(8): 2363-2377.
- [8] Zhang Y W, Zhou Y Q, Li X L, et al. Bas-Relief Generation and Shape Editing through Gradient-Based Mesh Deformation[J]. Visualization and Computer Graphics, IEEE Transactions on, 2015, 21(3): 328-338.
- [9] Schüller C, Panozzo D, Sorkine-Hornung O. Appearance-mimicking surfaces[J]. ACM Trans. Graph, 2014, 33(6): 216.
- [10] Arpa S, Süsstrunk S, Hersch R D. High Reliefs from 3D Scenes[C]//Computer Graphics Forum. 2015, 34(2): 253-263.
- [11] Luft T, Colditz C, Deussen O. Image enhancement by unsharp masking the depth buffer[M]. ACM, 2006.
- [12] Desbrun M, Meyer M, Schröder P, et al. Implicit fairing of irregular meshes using diffusion and curvature flow[C]//Proceedings of the 26th annual conference on Computer graphics and interactive techniques. ACM Press/Addison-Wesley Publishing Co., 1999: 317-324.
- [13] Fattal R, Lischinski D, Werman M. Gradient domain high dynamic range compression[C]//ACM Transactions on Graphics (TOG). ACM, 2002, 21(3): 249-256.