Cover Letter

We are thankful for the time and efforts by reviewers in reviewing this manuscript and appreciate their insightful and constructive comments. We have carefully revised our paper to address the issues in the review and highlighted all changes in red color. Our response to the reviewers' comments are described below.

Response to comments from reviewer1

1. The technical contribution of this paper is not significant. The whole work seems to be completed while not so solid. For example, only several examples have been shown in the paper. The reviewer expects to see some examples with more structural diversity and complexity.

We have added more examples in Figure 9 and Figure 12. Although the automatically generated examples can also be quickly constructed in other ways, the new added examples in Figure 12 are quite complex. We also conducted a user study and the results demonstrate that our system is significantly useful for both designing cartons and fabricating cartons.

2. For "minimizing the proximity function based on projection operators", it is better to give the proximity function here.

We are sorry that we did not explain the relationship of our formulation with the reference paper [3]. We rewrote the constraint equations in a similar way with Bouaziz et al. [3] and directly cite their paper as "Bouaziz et al. [3] proposed an efficient optimization method that combines all these shape constraints together. We directly use their C++ library ShapeOp to solve our shape optimization problem once a shape modification operation is performed."
The detailed implementation is not included in the paper and can be find in the corresponding reference.

3. It is better to show the interface of the suggestive system.

Our user interface is shown in our complementary video to show a more complete process of carton modeling using our system.

4. It is better to perform a user study to investigate the effectiveness of the suggestive system.

We conducted a user study to examine the productivity of our system to construct the corresponding 3D model of the planar layout, and the guiding effectiveness of users fabricating the physical mockup. More details about the procedure and results can be found in the added section "Sec 5.2 User Study" highlighted in blue.

5. In the related work section, it is better to give more interpretation about the relationship between Paper folding and the problems of Reconstruction from single line drawings and Shape

Optimization.

Yes. We added the relationship between our carton folding problem and line drawing problem in Section 2.2."Although having the same goal of constructing 3D models from 2D layouts including vertexes and edges, there is a one-to-one correspondence between the vertexes in the 2D line drawing and 3D model, while our problem recovers many-to-one correspondences and the 3D construction process is different."

Meanwhile, we added the relationship between our carton folding problem and shape optimization in Section 2.3, "The studies mentioned above focus on optimizing existing 3D shapes with geometric constraints, and propose different optimization solutions. 2D layouts are not involved in their problems. In contrast, our goal is to recover the structure and topology from an expanded layout to form a 3D carton model. The geometric constraints are automatically detected and presented for users to explore during the folding process in our system."

6. Paper folding is not an extensively studied problem. To make the paper understandable, it is better to clarify some terms such as Paste face (or give some reference resources).

We modified the annotation "paste face" to "glue flap", which sticks to the naming rule usually used by designers. We have also added the explaination of glue flap as "to ensure the carton is stable without any manual intervention. Most of these cases occur for attaching a glue flap to a side panel to make a closed box in practice.".

An illustration is also highlighted in Figure 5 to make sure the audience are clear about the concept of glue flap.

7. The visualization of the results need some improvements. Showing only one single-view colourful resulting folded box for each input is not a good visualization. Showing Multi-view and transparent results may be better.

We have added multi-view illustration of our generated models in Figure 9 and Figure 12 for visualization. As for the transparent mode, we tried to render our results with transparency. However, it was a bit messy because a large number of edges become visible in the figures.

Response to comments from reviewer2

1. References

#1 Martin Kilian, Simon Flöry, Zhonggui Chen, Niloy J. Mitra, Alla Sheffer, and Helmut Pottmann. 2008. Curved folding. ACM Trans. Graph. 27, 3, Article 75 (August 2008).

-#2 Martin Kilian, Aron Monszpart, and Niloy J. Mitra. 2017. String Actuated Curved Folded Surfaces. ACM Trans. Graph. 36, 3, Article 25 (May 2017).

#3 Solomon, J., Vouga, E., Wardetzky, M. and Grinspun, E. (2012), Flexible Developable Surfaces. Computer Graphics Forum, 31: 1567–1576

Carton folding is closely related to Origami, for which there is an extensive range of existing *research. For example:*

-#4 Tachi, T., 2009. Simulation of rigid origami. Origami, 4, pp.175-187.

-#5 Tachi, T., 2011. Rigid-foldable thick origami. Origami, 5, pp.253-264.

-#6 Tachi, T., 2010, November. Geometric considerations for the design of rigid origami structures. In Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium (Vol. 12, No. 10, pp. 458-460).

Thanks for pointing these reference to us. We have added the references and discussion in the Section 2.1 in the revised manuscript. "#1 presented an optimization-based framework to reconstruct a 2D development from a reference 3D surface. #2 introduced a discrete paradigm to model developable surfaces based on 2D configurations, where the crease pattern and the corresponding crease angles are pre-defined. #3 designed practical mechanisms to fabricate a curved folded surfaces simply by pulling a network of strings. By contrast, we focus on an inverse problem of recovering the 3D shape from a 2D planar pattern without knowing any 3D information. We also mentioned #4 #5 #6 in the folding motion of rigid origami problem and thick origami problem. These simulation systems are limited to the flat-foldable rigid origami with known mountain or valley crease types. In comparison, our system only takes the 2D crease pattern as input without knowing the folding angles of the creases."

2. I do not understand in what scenarios would a user need a computer algorithm to help with carton folding. As most carton patterns are designed with a target 3D shapes in mind, the folding instructions should have been generated during the design process. In addition, many of the carton layouts shown in this paper is quite trivial and intuitive, and it seems not necessary to use a computer to figure out how to fold them. The only case I can see where such a technique is helpful is a carton puzzle where the layout becomes rather complicated and there is no folding instructions available. But this seems rather niche. I think the authors should provide a more clear use case for this technique.

We added more background and applications in the Introduction. "Designers typically start package design by generating 2D vector artwork, then a 3D mockup is essential to be made for designers and clients to see the real appearance [8]. There are two ways to create a 3D mockup, a digital mockup is a great way of showing how the design would appear with lower cost and is convenient for designers to modify the layout repeatedly. A physical mockup is also helpful when making sure the size is correct and making a final decision. Due to the low cost and efficiency of digital mockups, there have been many software packages developed to help designers imp rove their design efficiency and productivity. For example, a commercial software named STUDIO [7] can generate 3D models by manually assigning angles to folding edges. With the model, users can turn their ideas into beautiful 3D images. Moreover, it is non-trivial for non-expert users to figure out how to fold a complex carton from a 2D crease pattern only. A digital simulation system will provide valuable references to inspire user creativities [29, 13, 20]. However, the existing softwares still need a lot of manual work to design cartons. In order to create a virtual 3D model and explore the layouts' diversity, our idea is to fold the existing layout directly into a 3D model, and reach a variety of layouts by manipulating the corresponding model."

3. The herustic of approximately 90 degree folding works well for box-like structures, but may not be able to take into account other folding scenarios. For examples, some carton layout involves folding of 180 degrees around an edge to produced a strong enforced face. The heuristic may not be able to capture such case. The authors should provide more discussion on the limitation of the heuristics and maybe present some failure cases.

We have added more discussion on the limitation in the Sec 5 and show a figure for the failure case.

Response to comments from reviewer3

1. The writing contains quite a few grammar errors, though fortunately in most cases these do not obstruct understanding of the method. Still, I recommend a thorough editing pass to clean up the exposition.

Sorry for the grammar errors. We have revised our paper carefully.

2. Is an angle of 90 degrees really the best to assume at the start? For example, a box that rolls up into a prism shape with many sides will need angles much less than 90 degrees, and folding each edge to the full 90 degrees might create a visual mess that is difficult for the user to interact with further. Maybe it makes more sense to start bending each crease up to a maximum of 90 degrees, but stopping if two parts of the box would ever interpenetrate?

The initialization is good for most cases in carton construction. There exist some cases where initialization may seem unreasonable. Adding some simple policy to initialization is a way to improve the initialized result. However, it still cannot cover all cases. Take the carton shown in Figure 12(f) as an example, if the initialization stops at an early step, the initialization will be like a pentagonal pyramid which confuses users, and the automatic detection may not work well as the vertices need to be merged are far away.

3. "Face pasting" mentions that in the case that constraints are conflicting, some of them are replaced by soft constraints. Which one? What are the details of the approach?

We have changed the explanation when soft constraints added to our algorithm. “Typically only a few shape modification operations are selected at each time, the above constraints form an under-constrained system to solve the new vertex coordinates. In order to find a feasible solution for the constrained problem, soft constraints will be added to keep the original positions of the vertexes that are not relative to the selected shape modification.” In the experiment, the soft constraints are always a part of constraints without replacing any other constraint.

4. In the vertex merging step, nearby vertices are sought, and then checked for compatible adjacent edges. Why not directly look for edges that have equal length and are nearby? This might catch mergable faces too.

The constraint used in our paper is to prescribe the shape based on a set of vertexes, and the input of optimization is a set of points. As a result, the automatic detection of vertexes is more intuitive for our algorithm. Edges that have equal length and are nearby should keep their endpoints to be merged separately in the implementation, and this operation is still based on vertexes in the end.

5. I don't understand the second condition in "Face pasting": won't (v_{ai} - v_c)\cdot n) always equal 0? Why is this check needed?

The second condition in "Panel pasting" is defined as

$$|(v_{ai} - v_c^b) \cdot n_b| < \epsilon_d,$$

where v_{ai} is ith vertex in Panel P_a, and v_c^b is the centroid of face P_b, and n_b is the normal of face P_b. This condition ensures the distance between two faces is under a threshold ϵ_d.

6. Section 5.2 hints at a tool for 3D editing, with the 3D changes mapped back to 2D. What precisely are these editing operations?

We describe the editing operation in Section 4.2. The system allows users to select an edge, and move it to the desired position to edit the 3D model. The editing operation is also shown in the complementary video from 00:19.