

Automatic Content-aware Projection for 360° Videos

Yeong Won Kim^{1,*†} Chang-Ryeol Lee^{2,*} Dae-Yong Cho² Yong Hoon Kwon²
 Hyeok-Jae Choi² Kuk-Jin Yoon²

¹VUNO, South Korea

ywon.kim@vuno.co

²Gwangju Institute of Science and Technology (GIST), South Korea

{crlee, dycho, hyeokjae94, yhkwon, kjyoon}@gist.ac.kr

Abstract

To watch 360° videos on normal 2D displays, we need to project the selected part of the 360° image onto the 2D display plane. In this paper, we propose a fully-automated framework for generating content-aware 2D normal-view perspective videos from 360° videos. Especially, we focus on the projection step preserving important image contents and reducing image distortion. Basically, our projection method is based on Pannini projection model. At first, the salient contents such as linear structures and salient regions in the image are preserved by optimizing the single Panini projection model. Then, the multiple Panini projection models at salient regions are interpolated to suppress image distortion globally. Finally, the temporal consistency for image projection is enforced for producing temporally stable normal-view videos. Our proposed projection method does not require any user-interaction and is much faster than previous content-preserving methods. It can be applied to not only images but also videos taking the temporal consistency of projection into account. Experiments on various 360° videos show the superiority of the proposed projection method quantitatively and qualitatively.

1. Introduction

Unlike traditional cameras which have a limited field of view (FOV), 360° cameras take omni-directional images at once. Therefore, it becomes much easier to capture the objects of interest or meaningful events, whereas the traditional cameras require careful viewpoint control. Recently, low-cost 360° cameras have been released thanks to the advance of Micro-Electro-Mechanical System (MEMS) technology, and lots of 360° videos are available on content dis-

tribution sites such as Youtube and Facebook.

To watch the 360° videos on normal 2D displays, the spherical images are projected onto the 2D plane with a limited FOV. As the FOV becomes wider, the projected image includes more contents and it becomes more similar to human perception, however, the distortion of the image becomes larger, and vice versa. Therefore, it is required to minimize the subjective distortion of image contents while approaching the possible widest FOV.

In the last decade, wide-angle projection has been studied in computer vision and graphics community. Most of the previous works such as rectilinear, stereographic, Panini projection [6] are based on a single fixed projection model. Naturally, it has limitation for preserving important contents in the image. Furthermore, while the center of the projection is less distorted, border regions of an image become more distorted. To handle this problem, Carroll *et al.* [2] proposed to minimize the distortion of contents such as salient lines and regions through optimization techniques. However, it requires manual extraction of lines to be preserved and is time-consuming. Rectangling stereographic [3] is another contents-preserving projection with automatic line extraction. However, it does not consider objects of interest in an image, and hard constraint on linear structures can cause large distortion on salient contents.

In this paper, we propose a fast and fully-automated contents-preserving projection method for 360° videos. We exploit the Pannini projection model [6] as a baseline among many projection models, which has an advantage of preserving not only conformality but also vertical lines. In addition, we can easily control the behavior of the Panini projection by adjusting two parameters. We take the linear structures (*i.e.* lines) and salient regions in images and videos into account for contents-preserving projection which are very important to increase the subjective quality of projected images. To preserve image contents *locally* and *globally*, we locally apply multiple Pannini projection mod-

*These authors contributed equally to this work.

†This work was done when the author was in Gwangju Institute of Science and Technology (GIST).

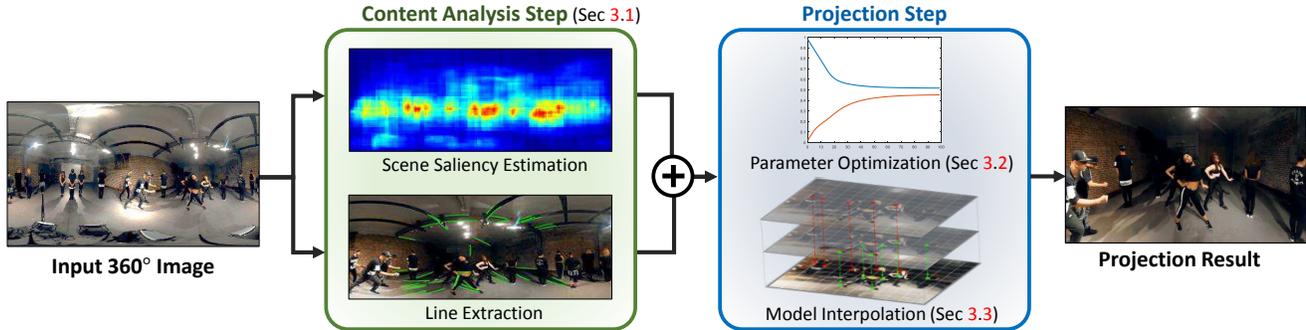


Figure 1. Overall framework for generating 2D normal-view videos from 360° videos. In this paper, although we cover the contents analysis step in Sec. 3.1, we mainly focus on the projection step. The contents analysis step such as viewpoint selection and salient region extraction can be replaced with any other methods.

els with different parameters to a single image in our framework where multiple parameters are adaptively optimized and spatially interpolated based on the image contents. Finally, we consider the temporal consistency of projection to generate temporally consistent and comfortable normal-view videos even under the severe viewpoint changes.

The main contributions of this paper are summarized as follows. First, we propose a contents-preserving projection method optimizing a single Pannini projection model. Second, we utilize multiple Pannini projection models to globally minimize contents distortion. Third, we enforce the temporal consistency for image projection to produce temporally stable normal-view videos.

The rest of this paper is organized as follows. In Sec. 2, we review various wide-angle projection methods. In Sec. 3, we describe the proposed methods to project the spherical image with less distortion, then show experiments on various 360 images and videos in Sec. 4. We finally conclude the paper in Sec. 5.

2. Related Works

Spherical image projection methods for wide FOV can be categorized into single-model based, multi-model based, and non-model based methods according to the number of models used for projection as follows.

Single-model based methods use a geometric model to project spherical panorama images onto an image plane. Rectilinear, Stereographic, and Pannini projection [6] models belong to this category. Rectilinear projection is the perspective projection from the center point of the spherical image onto the image plane. This model preserves all lines, but the contents on the margin of a projected image can be extremely stretched and distorted when the FOV is large. On the other hand, stereographic projection is the perspective projection from the opposite point to the point of tangency as the center of projection. This model preserves a conformality of the contents, but not lines. Pannini projection based on the cylindrical projection keeps the vertical

lines straight. Furthermore, the horizontal lines or radial lines are selectively preserved by the vertical compression. The methods mentioned above are very simple and fast, but have a common drawback — they can not preserve all lines and salient objects simultaneously.

Multi-model based methods project images with partially different multiple models depending on the contents such as lines and objects. They are comparatively simple and produce less distortion compared with the single-model based projection. However, they also yield strong distortion at the border of regions where different models are applied. Zelnik-Manor *et al.* [9] proposed a multi-model based method that applies locally different projections depending on scene structure in the panoramic images with user interaction. Rectangling stereographic [3] projects the spherical image onto a swung surface that is a combination of two orthogonal cylindrical projections with rounded edges. When the image is divided into four triangular regions with two diagonal lines, it respectively preserves vertical lines in left and right triangular regions and horizontal lines in upper and lower triangular regions of an image. However, it highly distorts the linear structures and objects straddling the diagonal lines.

Non-model based methods try to minimize projection distortion using optimization techniques [2]. Carroll *et al.* [2] proposed contents-preserving optimization-based projection method. It produces less distorted images than other approaches and well preserves important contents in the image. However, it is computationally much more expensive because of the iterative optimization process for every single point. Moreover, it requires non-trivial user interaction specifying straight lines to be preserved.

3. Proposed Framework

In this section, we present a fully-automated contents-aware projection from the 360° videos illustrated in Fig. 1. Here, note that, although the proposed framework includes the contents analysis step, we mainly focus on the projec-

tion step. The contents analysis steps such as viewpoint selection and salient region extraction can be replaced with any other methods. With given viewpoint, we project some portion of a spherical image onto the 2D image plane with less distortion. We exploit the Pannini projection model [6] as a baseline which has an advantage of preserving not only conformality but also vertical lines well. As in [11], we consider two properties, conformality of salient objects and curvature of linear structures, to measure distortions.

We first extract line segments and salient objects to be preserved automatically. Then, we define distortion measures for the Pannini projection model [6] and optimize the Pannini parameters to minimize defined distortions. If we have multiple salient objects in an image, we compute multiple optimal Pannini projection parameters for multiple salient objects, respectively. Afterwards, we perform the model interpolation to minimize contents distortion globally and locally. Since the optimization is performed for a few number of model parameters (two for Pannini projection), our method is much faster than other optimization-based methods. Furthermore, we also consider the temporal consistency of the projection to generate temporally consistent and comfortable perspective videos even under severe viewpoint changes.

Algorithm 1 Proposed Contents-Aware Projection

- 1: Input: $I_{1:N}^s$, 360 spherical image sequence.
 - 2: Output: $I_{1:N}^n$, normal-view image sequence.
 - 3: **for** $I_t^s \in I_{1:N}^s$ **do**
 - 4: Extract L_t, P_t from I_t^s in Sec. 3.1.
 - 5: Compute $(d_t^g, w_t^g)^-$ using (L_t, P_t) by Eq. (8)
 - 6: Compute $(d_t^g, w_t^g)^+$ from $((d_t^g, w_t^g)^-, (d_{t-1}^g, w_{t-1}^g)^+)$ by Eq. (9)
 - 7: **for** $p_i \in P_t$ **do**
 - 8: Get L_n^i, P_n^i , subsets of L_t and P_t around p_i
 - 9: Compute $(d_t^i, w_t^i)^-$ using (L_n^i, P_n^i) .
 - 10: Compute $(d_t^i, w_t^i)^+$ from $((d_t^i, w_t^i)^-, (d_{t-1}^i, w_{t-1}^i)^+)$.
 - 11: Compute $Proj_t^-$ from $(d_t^g, w_t^g, d_t^i, w_t^i, P_t)$ by Eq. (6-7).
 - 12: Compute $Proj_t^+$ from $(Proj_t^-, Proj_{t-1}^+)$ by Eq. (10).
 - 13: Get I_t^n from I_t^s using $Proj_t^+$.
-

3.1. Image Content Analysis

In this paper, we take the linear structures (*i.e.* lines) and salient regions in images and videos into account for contents-preserving projection which is very important to increase the subjective quality of projected images. Note that the proposed projection method is not dependent on the choice of the image content analysis methods.

To find linear structures in a spherical image, we exploit an advantage of the rectilinear projection which preserves every line in the image.¹ We project a partial spherical

¹Any methods detecting lines on spherical images can be applied.

image with rectilinear projection and then extract line segments using Line Segment Detector (LSD) [7] from the projected image. Each line segment is transformed from image coordinates to spherical coordinates. In general, distortions of short line segments are less perceivable, so only the line segments longer than a pre-defined threshold are used.

To extract salient objects, we compute scene saliency as the combination of appearance and motion saliency of the image as

$$S_i^{scene} = wS_i^{appear} + (1 - w)S_i^{motion}, \quad (1)$$

where S_i^{scene} , S_i^{appear} , and S_i^{motion} denote scene saliency map, appearance saliency map, and motion saliency map of the partial image, i . w is a weight parameter.

To find salient objects, we define the appearance saliency as a probability of object existence in the image. Therefore, we exploit objectness-based object proposals [1, 4, 10] that generate multiple bounding boxes with objectness scores. The objectness score presents how likely the bounding box contains an object. We estimate the appearance saliency map by accumulating the objectness score of each object proposal. To estimate motion saliency S^{motion} , we exploit the method proposed in [5] with optical flow [8] as an input.

We assume that objects have higher scene saliency than the background. Thus, we extract local peaks as salient objects by applying non-maximum suppression to the scene saliency. Note that any other saliency detection methods can be applied to our projection method.

3.2. Optimal Pannini Parameter Estimation

To preserve the extracted linear structures and salient points, we use the Pannini projection model as a baseline because it can selectively preserve contents by changing parameters. The Pannini projection model is defined as

$$u_p = \frac{(d + 1) \sin(\phi_p)}{d + \cos(\phi_p)},$$

$$v_p = \tan(\theta_p) \left(\frac{(d + 1)(1 - w)}{d + \cos(\phi_p)} + \frac{w}{\cos(\phi_p)} \right), \quad (2)$$

where θ_p and ϕ_p denote a point on spherical coordinates, and u_p and v_p denote a correspondence of θ_p and ϕ_p on the image coordinates. d and w are control parameters. d is a distance between the projection plane and the center of projection. If d is equal to 0, the projection becomes rectilinear projection which preserves linear structures but stretches the boundary of perspective images. If d is equal to 1, it is cylindrical stereographic projection which preserves shape of objects and vertical linear structures but bends radial linear structures. w is a weighting parameter for vertical compression, which makes horizontal linear structures straight but distorts radial linear structures. Therefore, optimal parameters should be determined depending on contents.

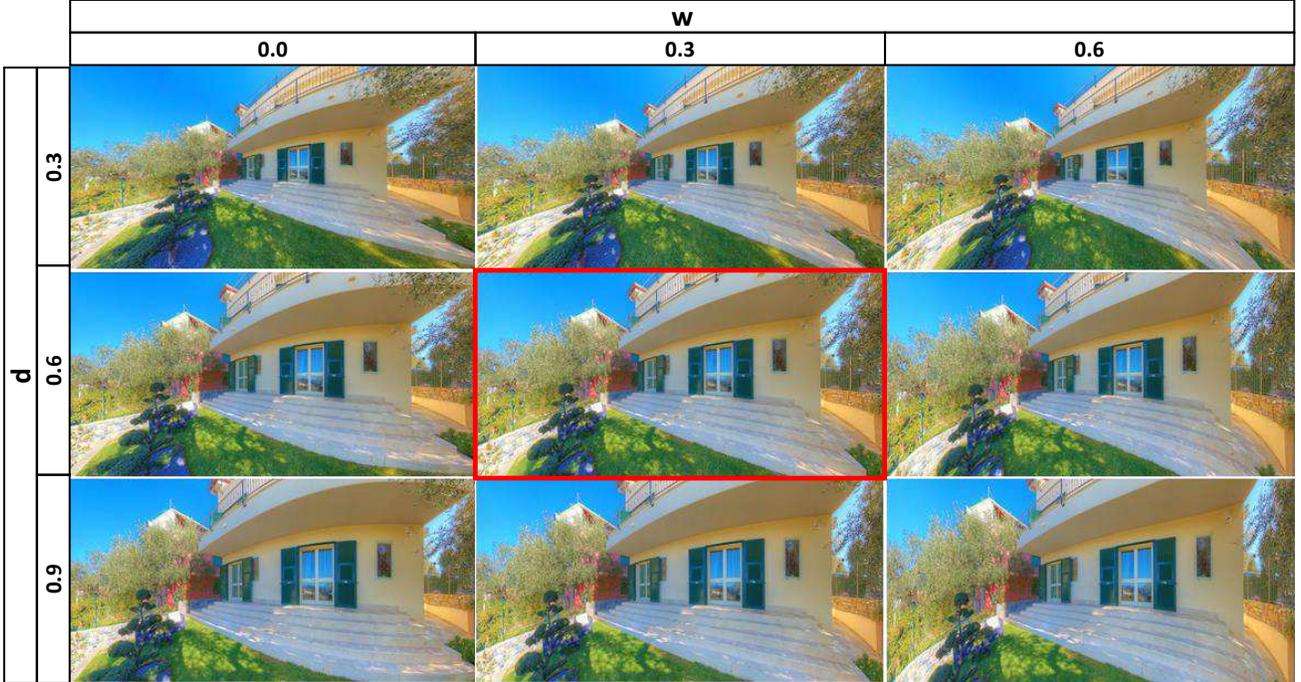


Figure 2. Estimation of optimal Pannini parameter d and w . The red box denotes the Pannini projection result with optimized parameters.

To estimate optimal parameters to preserve both linear structure and salient objects, we define two distortion measures as illustrated in Fig. 4. To consider the straightness of linear structure, we define a distortion measure as a distance between the middle point of the line segment and the line which is defined by two endpoints of the line segment on the image plane after projection. It is formulated as

$$D(l) = \left(\frac{u_s(v_e - v_m) + u_e(v_m - v_s) + u_m(v_s - v_e)}{\sqrt{(u_s - u_e)^2 + (v_s - v_e)^2}} \right)^2, \quad (3)$$

where subscript s , m , and e denote the starting, middle, and end point of the line segment, l , respectively. If the line is bent when it is projected, the measure has a high value.

To consider shapes of salient objects, we adopt the distortion measure of [2] which is defined as

$$C(p) = \left(\cos\theta_p \frac{\Delta u_p}{\Delta\theta_p} + \frac{\Delta u_p}{\Delta\phi_p} \right)^2 + \left(\cos\theta_p \frac{\Delta v_p}{\Delta\theta_p} - \frac{\Delta u_p}{\Delta\phi_p} \right)^2. \quad (4)$$

This measure presents a conformality of a point p . With the two measures, we define an objective function as

$$E(L_t, P_t) = \omega_d \sum_{l \in L_t} D(l) + \omega_c \sum_{p \in P_t} C(p), \quad (5)$$

where L_t is a set of line segments and P_t is a set of salient points at frame t . ω_d and ω_c are weighting parameters that determine which components are more preserved. The objective function is minimized by the steepest decent method.

Because it has only two parameters, it is very fast. This optimization is globally applied to consider every linear structure and salient object in the image simultaneously. However, it cannot preserve all components simultaneously because it uses a single model to the whole image. Figure 2 shows several projection results with various values of the parameters. The parameters, which are obtained by the proposed optimization method, shows the best results.

3.3. Model Interpolation

To cover remaining distortions, we adopt a multi-model based approach proposed by Zelnik-Manor *et al.* [9]. The main observation is that the centers of the projected images are less distorted. Thus, for one salient point, we project an image around the salient point with a model of which viewpoint is centered at the salient point. Then, shapes around salient points are preserved. However, regions between salient points have strong distortions because projection models are different from each other. To reduce these distortions, we spatially align multiple models in an image.

First, we set the Pannini projection model with globally optimized parameters as a global model. The global model determines the whole structure of a perspective image and locations of local models that are projections of salient points. In this process, if equivalent objects are projected on different locations by the global model and the local models, distortions of these objects should increase in the final results. Thus, we applied a transition process to and scaled the local models to match the center of each lo-

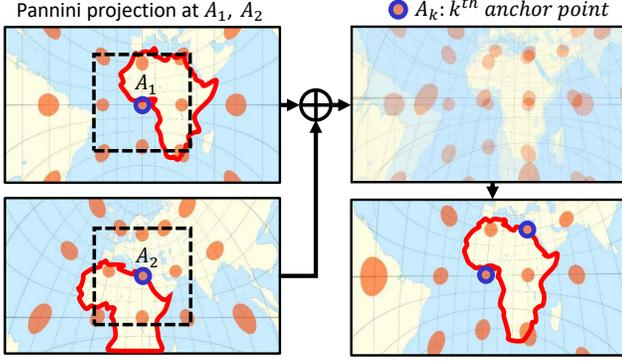


Figure 3. The concept of model interpolation.

cal model to the global model. To do this, we define anchor points as the salient points projected by the global model. When center points of local models locate on anchor points, shapes projected by local models are aligned with shapes projected by the global model.

After the alignment of the local models, we interpolate local models to fill the regions between salient points smoothly. The interpolated model is defined as

$$Proj(u, v) = \frac{w_g(u, v)}{Z} Proj_g(u, v) + \sum_k \frac{w_{A_k}(u, v)}{Z} Proj_{A_k}(u, v), \quad (6)$$

$$w_P(u, v) = c_P e^{-\frac{d_P(u, v)^2}{2\sigma}}. \quad (7)$$

$Proj_P$ is the Pannini projection model with P as the center of the model. It is a backward projection that transforms points on UV coordinates to spherical coordinates. Z is a normalizing factor. $d_P(u, v)$ denotes Euclidean distance between a point (u, v) and a point P . g and A_k represent the center points of the global and the local projection models composing the interpolated projection model. They are subset of P in Eq. (7). Thus, $Proj_g$ represents the global model and $Proj_{A_k}$ represents the local model. Therefore, $\{w_g, w_{A_k}\} \in w_P$ are control parameters to decide the weights of the global and the local models. w_g, w_{A_k} have their own parameter c , that is, c_g, c_{A_k} .

To preserve shapes around salient points, weight $w_P(u, v)$ is defined in an exponential form decreasing according to $d_P(u, v)$. Then, the region nearby an anchor point A_k is substantially influenced by $Proj_{A_k}$ and projected by $Proj_{A_k}$. On the other hand, a distant region from A_k is almost unaffected. Therefore, the interpolated model changes smoothly to another projection models. Fig. 3 illustrates the concept of the model interpolation. With two anchor points, A_1 and A_2 , $Proj_{A_1}$ and $Proj_{A_2}$ are determined as shown in the left side of the figure. Then, they are aligned and merged to generate the interpolated model.

3.4. Temporal Consistency for Video Projection

The projection model could fluctuate when line segment, salient points, or viewpoints are changed frequently. To eliminate the fluctuation for video projection, we enforce the temporal consistency in projection. First, we make parameters of the Pannini projection model consistent temporally. Because parameters determine the behavior of the projection model, a little change of parameters can make severe fluctuation.

To handle this, we add the penalty term $E_{penalty}$ to the objective function for optimization to smooth parameters as

$$arg \max_{d_k, w_k} E(L_t, P_t) + E_{penalty},$$

$$where E_{penalty} = \omega_{pd}(d_t - d_{t-1})^2 + \omega_{ps}(w_t - w_{t-1})^2. \quad (8)$$

Here, $E(L_t, P_t)$ is the objective function in Eq. (5). d_t, w_t indicate the Pannini parameters at frame t , and ω_d and ω_s are weighting parameters. Eq. (8) enforces the estimated parameters in the current frame to be similar with the parameters of the previous frame.

Furthermore, to make the change of the parameters smoother, we apply the exponential moving average on the parameters estimated from Eq. (8) as

$$d'_t = \omega_{md}d_t + (1 - \omega_{md})d'_{t-1}, \\ w'_t = \omega_{ms}w_t + (1 - \omega_{ms})w'_{t-1}, \quad (9)$$

where $\omega_{md} \in [0, 1]$ and $\omega_{ms} \in [0, 1]$ indicate weighting parameters.

Finally, we use the exponential moving average pixel-wisely to the interpolated model $Proj$ which is defined in Sec. 3.3. It is defined as

$$Proj^t(u, v) \Leftarrow \omega_p Proj^t(u, v) + (1 - \omega_p) Proj^{t-1}(u, v), \quad (10)$$

where $\omega_p \in [0, 1]$ is a weighting parameter. $Proj$ maps a point at (u, v) on the perspective image to the point at (ϕ, θ) on the spherical image. Furthermore, we adaptively adjust the weighting parameter ω_p according to change of viewpoint. For example, when the viewpoint remains stationary, inconsistently estimated Pannini projection parameters can cause discomforts to viewers. Otherwise, when the viewpoint changes, fluctuations due to inconsistently estimated Pannini projection parameters are less noticeable since contents in the perspective image change rapidly. Therefore, we increase the weighting parameter ω_p when the viewpoint remains stationary, and we decrease ω_p when the viewpoint changes.

4. Experimental Results

In this section, we compare our projection method with the Pannini projection model [6] which is the baseline of

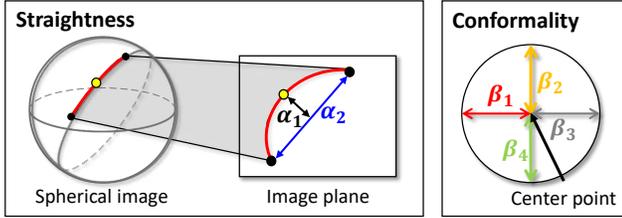


Figure 4. Quantitative evaluation measures

our algorithm, and other state-of-the art projection methods: Carroll *et al.* [2] and Rectangling Stereographic projection [3]. The 360 ° image and video datasets for our experiments are collected from the web.

In our experiments, we set $c_g, c_{A_k}, \omega_d, \omega_c, \omega_{pd}, \omega_{ps}, \omega_{md}$ and ω_{ms} to 2.0, 1.0, 10^{-3} , 10^{-4} , 0.999, 10^{-6} , 0.9, and 0.9, respectively. ω_p is set to 0.99 when the viewpoints move. Otherwise, ω_s is set to 0.8. Horizontal FOV and aspect ratio is set to 150° and 16:9 or 170° and 21:9.

4.1. Quantitative Evaluation

For quantitative comparisons of the proposed method with other methods, we introduce two distortion measures: straightness and conformality measures. The straightness measure indicates the degree in which a line segment in the real-world is bent on the projected image. As shown in Fig. 4, given two endpoints of a line segment, we define the straightness measure as

$$Straightness = \frac{\alpha_2}{\alpha_1 + \alpha_2}, \quad (11)$$

where α_2 is the distance between the two endpoints and α_1 is the perpendicular distance between the middle point of the curved line segment and a line that joins the two projected endpoints. The straightness measure is close to 1 if the distortion of the line segment is low, otherwise 0. As the second measure, we consider the conformality, *i.e.*, measuring the degree to which the appearance of an original spherical image is distorted around salient points. To measure the conformality, we sample four points around a salient point. The points are extracted at the spherical image coordinates moved by 0.1 *rad* in pitch or roll direction. Then, when the four points are projected on the 2D image plane as in Fig. 4, the conformality measure is defined as

$$Conformality = \frac{\min(\beta_1, \beta_2, \beta_3, \beta_4)}{\max(\beta_1, \beta_2, \beta_3, \beta_4)}, \quad (12)$$

where β_n is a distance between the salient point and the projected sampled point. If the four values from β_1 to β_4 are similar to each other, this value is close to 1, which means that the shape around the salient point is less distorted.

We compared the proposed method with the rectilinear and Pannini projection methods. To exclude dependency on

Table 1. Quantitative evaluation of projection results. The best and second best scores in each rank are marked with red and blue. Straightness and conformality scores are respectively averaged on 11 lines and 16 points.

Algorithm	Straightness		Conformality	
	Average		Average	
	16:9	21:9	16:9	21:9
Rectilinear	0.9999	0.9999	0.5235	0.4735
Pannini (d=1.0)	0.9639	0.9747	0.7534	0.8763
Pannini (d=0.5)	0.9748	0.9825	0.7506	0.7298
Optimized Pannini	0.9995	0.9999	0.5848	0.6011
Proposed method	0.9705	0.9874	0.7539	0.8407

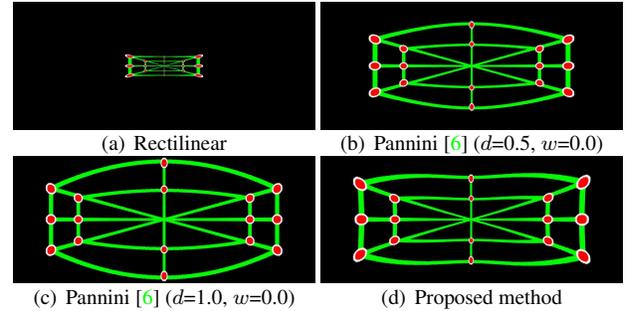


Figure 5. Results of our projection model and other projection models with synthetic input. (FOV = 170°, Aspect ratio = 21:9)

the content analysis step in Sec. 3.1, we used manually extracted salient points and line segments as input. For quantitative evaluation, we generated synthetic spherical images and then projected them using projection methods with the aspect ratio of 16:9 with the FOV of 150°, or with the aspect ratio 21:9 with the FOV of 170°. We used parameters (d=1.0,w=0) for Pannini algorithm, that is commonly used. Additionally, we included Pannini projection images with d = 0.5 for a variety of comparisons. Fig. 5 show the results of the projected images, and the red points and green lines represent salient points and line segments, respectively. Table 1 represents the results of measuring the straightness and the conformality for each method. The rectilinear projection obtains the highest straightness score but the lowest conformality score, whereas the Pannini method (d=0.68) yields high conformality score but low straightness score. Our method achieves high scores in both straightness and conformality. It demonstrates that our method maintains both the straightness and conformality highly in every projection environment. Especially, the minimum straightness scores of our method are higher than other methods except that of the rectilinear method. It means that the proposed method guarantees the competent quality for the straightness compared to the other methods.

4.2. Subjective Evaluation with User Study

We perform a user study to verify whether the results of the proposed projection method are comfortable or not with help from the crowd sourcing service *Amazon Mechan-*

Table 2. Subjective evaluation results. Votes on the 21 sets are averaged, and the range of the average votes is [0, 100]. The proposed methods without and with the model interpolation are denoted as Optimized Pannini and Proposed, respectively. The best and second best scores are marked with red and blue.

	Rectilinear	Stereographic	Pannini [6]		Carroll [2]	Rect. Stereographic [3]	Optimized Pannini	Proposed
			d=1.0	d=0.5				
Average votes	19.57	37.24	39.43	42.76	40.38	37.23	37.62	45.05

Table 3. Computational time of the proposed method.(second per frame)

Salient point extraction	Line extraction	Pannini parameter optimization	Model interpolation	Total
0.466	0.035	0.001	0.240	0.742

ical Turk. We generated 21 sets with 8 projection methods: rectilinear, stereographic, Pannini, Carroll, rectangling stereographic, the proposed method without and with model interpolation. With these 21 x 8 images, we performed blind test with 100 subjects. For each set, we showed 8 projection results in a random order, and asked a question, "Choose the best 3 photos among below 8 photos", to the 100 subjects. The votes on each projection method over 21 sets were averaged. As shown in the Table 2, the proposed method with model interpolation produces the most preferred results on average. However, Pannini got higher scores than optimized Pannini. The proposed method has two-step strategy to project still images. The optimized Pannini is the first step of the proposed method. In the first step, we focus on preserving the straightness of lines as possible. Then, preserving conformality is emphasized in model interpolation stage. For this reason, the optimized Pannini preserves straightness well but does not conformality. Unfortunately, excessively preserved straightness can cause discomfort to viewers as shown in the result on rectilinear projection, which completely preserves the linearity of lines in an image. Fig. 6 shows some results which were evaluated.

4.3. Qualitative Evaluation

We perform qualitative comparison for still images. Pannini projection result is generated with default parameters, ($d = 1.0, w = 0.0$), preserving the conformality as much as possible. The result of Carroll *et al.* [2] is obtained with automatically extracted line segments (the same as ours) as inputs. For the proposed method, automatically extracted line segments and salient points are also used as inputs.

As shown in Fig. 6, the Pannini projection [6] bends horizontal lines, whereas our results with model interpolation preserves both lines and objects. Our result with only parameter optimization shows that linearity and conformality are more preserved than Pannini. The result of the rectangling stereographic projection looks similar with our result with model interpolation, but lines at the boundaries of the image are bent severely because they are on the border of the different models. Carroll *et al.* [2] do not preserve the

linearity at some lines and end of the lines because the extracted lines are fragmented and missed. More qualitative comparison results are included in our supplementary material.

4.4. Results for 360° Videos

We test our method with three 360° videos. Spherical image sequences and viewpoint trajectories are provided as inputs. The results of the proposed method are shown in Fig. 7 where the top and bottom images for each dataset represent the spherical image and the projection image, respectively. Green regions in the upper images denote the projected area. We observe that the proposed projection method reduces the distortion around the contents such as straight indoor structures and faces of people. Furthermore, it provides temporally consistent image sequences, which is verified through the video in the supplementary material.

4.5. Computational time

Table 3 shows computational time for each step of the proposed method with a single core of 4.00GHz. The salient point extraction spends over half of the total computational time in our framework. However, it can be improved by substituting it with faster saliency detection algorithms. Also, the model interpolation takes one-third of the total computational time but it can be reduced by parallelization. The proposed optimization method is much faster than other optimization-based methods. It takes less than 0.001sec (when implemented in C++), whereas Carroll [2] takes about 2sec with GPU (implemented in Photoshop) on the same PC.

5. Conclusion

In this paper, we have presented a fully-automated content-aware projection for 360° videos. To this end, we proposed multi-model based Pannini projection optimization method that preserves both linear structures and salient objects which are automatically extracted. Additionally, we considered temporal consistency to generate temporally stable videos. Experiments including user study show that the proposed projection method is much faster and produces better results than previous content-preserving methods on various environments.

Acknowledgment. This work was supported by the Visual Display Business (RAK0117ZZ-21RF) and the Samsung Research Funding Center (SRFC-TC1603-05) of Samsung Electronics.

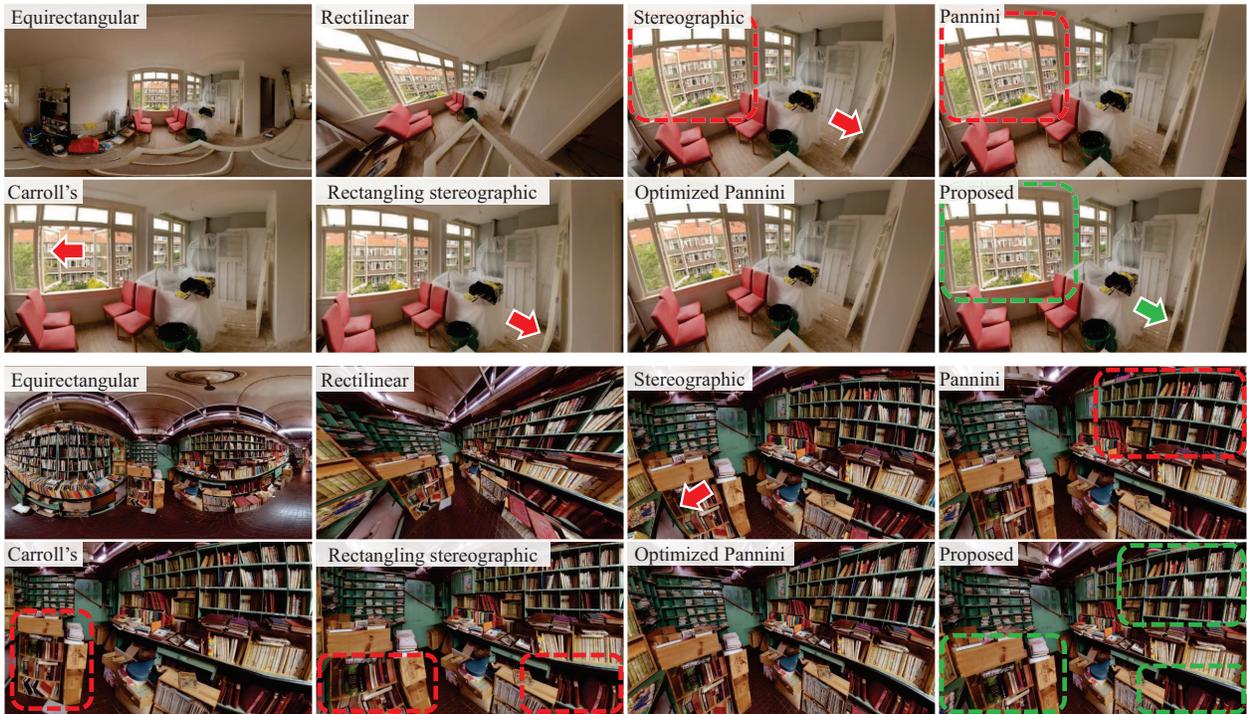


Figure 6. Some results evaluated on *Amazon Mechanical Turk*

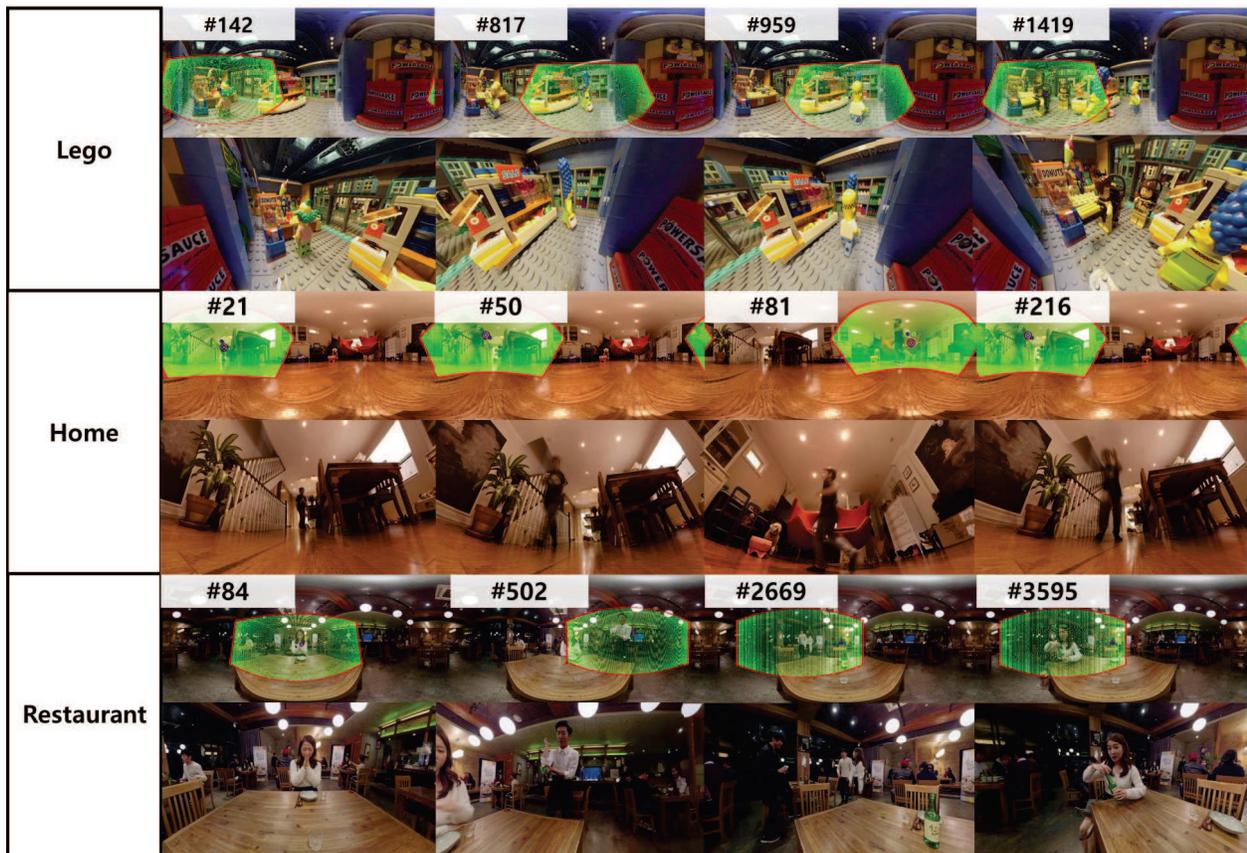


Figure 7. Results of the proposed method for three 360° videos. The proposed method produces temporally consistent and content-preserving projection results from the 360° videos with the given viewpoints.

References

- [1] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2189–2202, 2012. 3
- [2] R. Carroll, M. Agrawala, and A. Agarwala. Optimizing content-preserving projections for wide-angle images. *ACM Transactions on Graphics (TOG)*, 28(3):43, 2009. 1, 2, 4, 6, 7
- [3] C.-H. Chang, M.-C. Hu, W.-H. Cheng, and Y.-Y. Chuang. Rectangling stereographic projection for wide-angle image visualization. In *IEEE International Conference on Computer Vision (ICCV)*, 2013. 1, 2, 6, 7
- [4] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. H. S. Torr. BING: Binarized normed gradients for objectness estimation at 300fps. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 3
- [5] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *IEEE International Conference on Computer Vision (ICCV)*, 2007. 3
- [6] T. K. Sharpless, B. Postle, and D. M. German. Pannini: a new projection for rendering wide angle perspective images. In *Proceedings of the Sixth International Conference on Computational Aesthetics in Graphics, Visualization and Imaging*, 2010. 1, 2, 3, 5, 6, 7
- [7] R. G. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall. Lsd: a line segment detector. *Image Processing On Line*, 2:35–55, 2012. 3
- [8] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. Deepflow: Large displacement optical flow with deep matching. In *IEEE International Conference on Computer Vision (ICCV)*, 2013. 3
- [9] L. Zelnik-Manor, G. Peters, and P. Perona. Squaring the circle in panoramas. In *IEEE International Conference on Computer Vision (ICCV)*, 2005. 2, 4
- [10] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *European Conference on Computer Vision (ECCV)*, 2014. 3
- [11] D. Zorin and A. Barr. Correction of geometric perceptual distortions in pictures. In *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, 1995. 3