Learning Compressible 360° Video Isomers

Yu-Chuan Su Kristen Grauman
The University of Texas at Austin

The supplementary materials consist of:

A Details for the compression parameters
B Rotational symmetry analysis
C More qualitative examples
D More failure cases

Please refer to the our project webpage[1] for the video examples.

A. Compression Parameters

We encode the videos using x264/x265/libvpx through FFMPEG. The compression parameters of FFMPEG for each encoder are as follows.

- x264 — "-preset medium -crf 0 -an"
- x265 — "-preset medium -x265-params lossless=1 -crf 0 -an"
- libvpx — "-speed 4 -cpu-used 4 -lossless 1 -qmin 0 -qmax 0 -an"

The "-an" option disables audio in the output bit-stream. The "-preset medium" in x264/x264 and "-speed 4 -cpu-used 4" controls the encoding speed. We use the default setting for x264/x265 which provides a reasonable balance between speed and compression rate. Other options specify lossless compression for each encoder. For the transform360 filter, we use bicubic interpolation for pixel values.

B. Rotational Symmetry

We justify our design of restricting the rotation within $90°$. We compute the correlation between cubemap size related by $90°$ rotation along either θ or ϕ. In order to do so, we compute the size of cubemaps with $\{\pm 60°, \pm 75°, \pm 90°\}$ rotation along either pitch or yaw and compare them with those with rotation within $[-45°, 45°]$. The correlations are in Table 1. We also show the correlations for $45°$ rotation for comparison. The strong correlation clearly shows that the cubemap sizes are indeed symmetric to $90°$ rotation.

[1] http://vision.cs.utexas.edu/projects/360isomers

Encoders	H264	HEVC	VP9
90°	1.00	0.95	0.99
45°	0.25	0.23	0.23

Table 1: Correlation between cubemap sizes related by $90°$ and $45°$ rotation.

C. Qualitative Examples

In this section, we show more qualitative examples similar to Fig. 8 in the main paper. See Fig. 1 and 2. We can see that even small objects can affect the compression rate, such as the rhinos in the first example and the diver in the second example. The pattern doesn't even have to correspond to a real object like the blank region in the fourth example or the logo in the fifth example. The fifth example also shows how the file size is affected by multiple factors jointly. The distribution would be symmetric with respect to θ=0 if the file size only depended on the logo, but the sky and cloud lead to the additional mode at the top middle. We also see that the continuity of foreground objects is not the only factor that matters from the sixth and seventh example; the person in the sixth example and the pilot in the seventh example lie on the face boundary in the optimal orientation. The result suggests that heuristics based on object location, either automatic or manual, do not solve the problem. The eighth example shows that the compression would be more efficient if the motions fall in the same face even if it does not introduce discontinuity in motion.

D. Failure Cases

In this section, we show failure examples similar to Fig. 8 in the main paper. See Fig. 3. In the first example, we can see the best compression rate occurs when the coral is continuous, while our method fails because it decides to keep the sun light (round white pattern) continuous. In the second and third example, the video size tends to be smaller when the horizon falls on the face diagonal, possibly because it is more friendly for intra-prediction in compression. Our method doesn't learn this tendency, so it fails to predict the optimal θ and only predicts the correct ϕ.

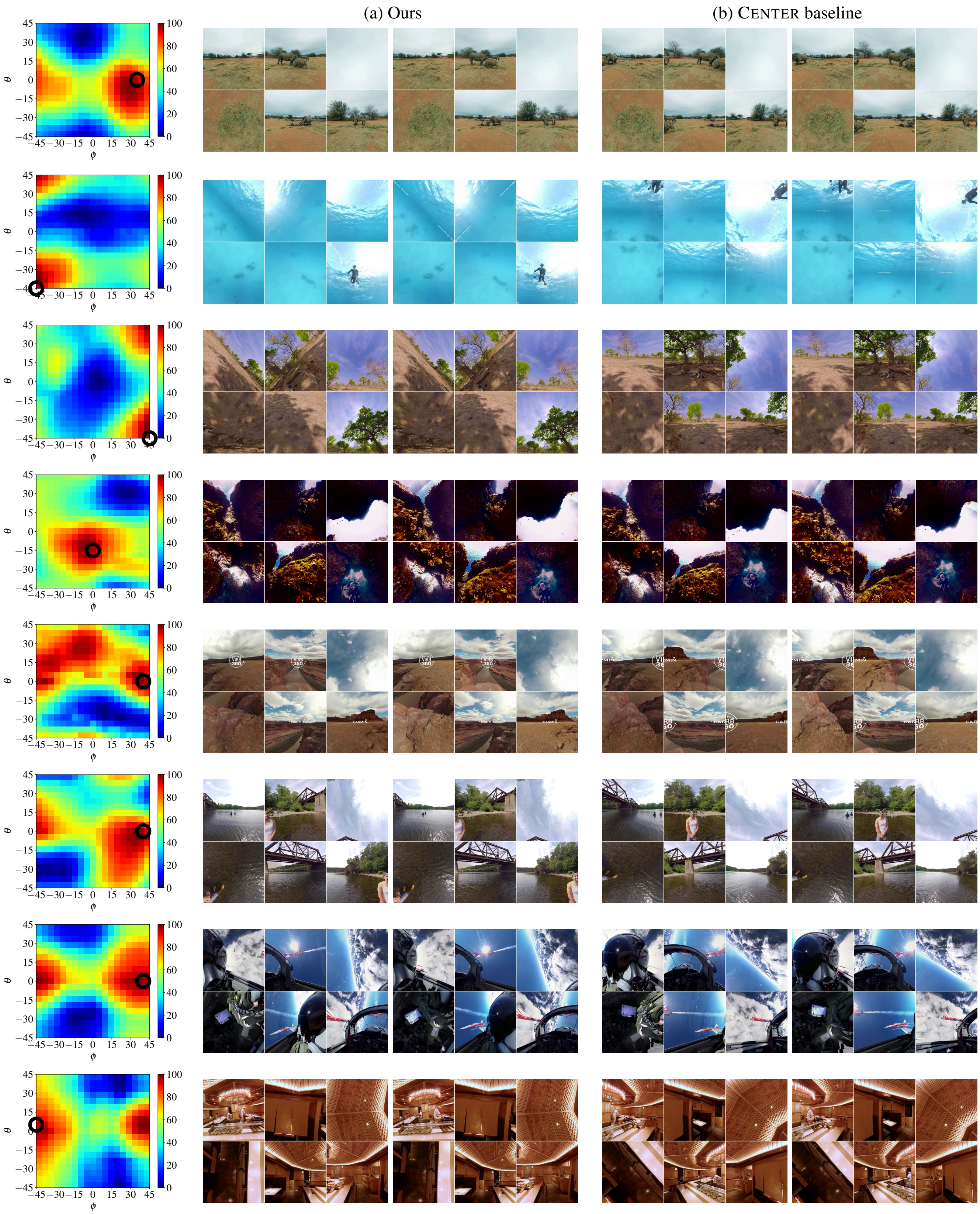

Figure 1: Qualitative results. Each row shows a clip. The first figure per row shows the size distribution. Black circle shows the predicted result, which is rendered in the second and third figures. The fourth and fifth figures show the CENTER baseline. The second and fourth figures show the first frame of the clip, and the third and fifth figures are the last frame.

Figure 2: Qualitative results (cont.).

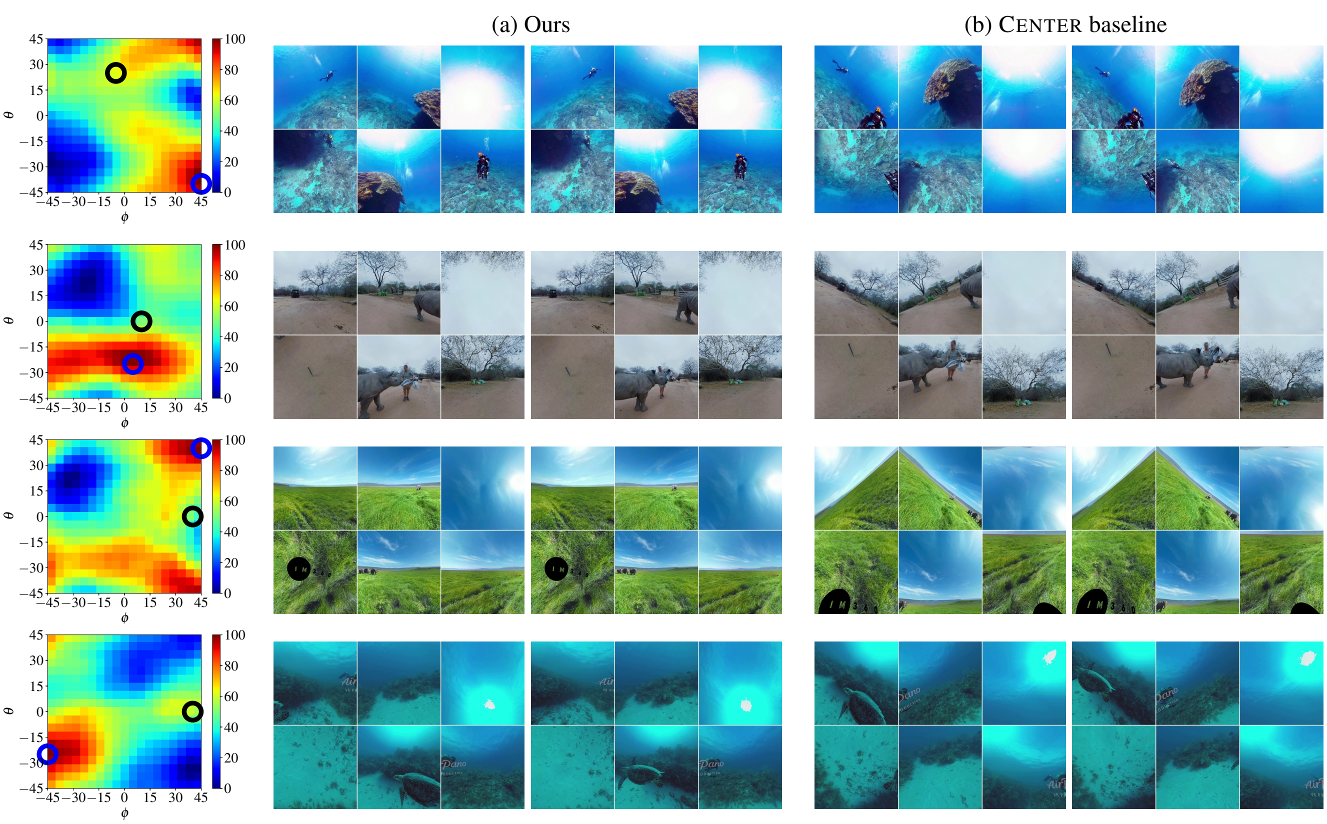

Figure 3: Failure cases. Blue circle shows true minimum and is rendered in the fourth and fifth figures. The two figures are the first and last frame of the clip.