The final published version is available at http://dx.doi.org/10.1109/ISM.2009.40.

Synthetic Video Geneartion with Complex Camera Motion Patterns to Evaluate Sprite Generation

Yi Chen
Computer Science Department
University of Alabama in Huntsville
Huntsville, United States
yichen@cs.uah.edu

Ramazan S.Aygün
Computer Science Department
University of Alabama in Huntsville
Huntsville, United States
raygun@cs.uah.edu

Abstract— **Without the ground truth image, there is no proper objective method to evaluate sprite generation. In this paper, we propose several complex camera motion patterns to generate synthetic video from original images. Our camera motion patterns include rotation, affine, and Pan-Tilt-Zoom (PTZ) transformations. In addition, our camera motion patterns also include combined patterns. Subsequently, we applied sprite generation to the synthetic videos. Objective evaluation is performed by comparing the ground truth image and sprite based on Peak-Signal-to-Noise Ratio (PSNR) and size. Pattern algebra file is also provided for estimating the accuracy of sprite generation about global motion parameters. Our result indicates that frame PSNR, picture PSNR, size, ground truth image, and pattern algebra file are good indicators of sprite quality.**

Keywords- sythetic video generation; sprite generation

I. INTRODUCTION

Sprite (or mosaic) generation has become significant with the introduction of the MPEG-4 Visual Standard [1]. MPEG-4 provides efficient video compression and interactivity with objects. MPEG-4 Main Profile is featured with sprite coding by applying motion compensation to eliminate the errors that come through sprite generation. For the latest standardized codec H.264/AVC[2] [3], it has been shown that the generation of a background sprite image containing all the background information of a certain sequence is effective for video compression.

However, the accuracy of sprite generation is not properly considered in MPEG-4 and H.264/AVC. The existing evaluation of generated sprite is composed of two phases. In the subjective evaluation, an expert judges the accuracy of the sprite by comparing the generated sprite and original video. In the objective evaluation, each frame of the original video is regenerated from the sprite using the motion parameters that are used in sprite generation and then the error between original frame and generated frame is computed. The objective method alone can not be sufficient for the accuracy of the sprite. For example, if the frames of a video are concatenated without alignment, the frame will could be generated 100% accuracy without the correct sprite.

To check the accuracy of sprite, it is better idea to use a ground truth image if the ground-truth is available. In the literature, there is some work on synthetic video generation from image or video to evaluate the performance of video processing.

Black and Ellis [4] generated ground truth tracks for objects and then embedded these into videos to observe the performance of video tracking algorithms with dynamic occlusions. In [5], "work-through" generates an image inside the scene and provides a realistic sensation by rotating a camera or using multiple cameras. In our early work [6], we used zigzag, spiral, earthquake, and zoom patterns to generate video from 2D images.

The metrics used for video processing is mainly related to object tracking rather than sprite generation. Without applying the ground truth, Erdem [7] used two measurements 1) color and motion difference around the boundary of the estimated video object plane and 2) the color histogram difference between the current object plane and its temporal neighbor. With ground truth, they present four objective metrics [8]: misclassification penalty, shape penalty, motion penalty, and combined penalty. For sprite generation [6], we used Peak-Signal-to-Noise Ratio (PSNR), sizes of images, and error on the estimation of motion as measurements. However, our previous work included only translational and zoom patterns. Although the sprites of translational patterns are pretty satisfactory, the sprites of the zoom pattern are not satisfactory.

Our motivation for the work presented in this paper is to include more complex camera motion patterns to check the performance of sprite generation. At the same time, the global motion parameters for each frame are collected in pattern algebra file, and ground truth images are presented to be used for measuring accuracy.

The remainder of this paper is organized as follows. Section 2 describes the camera motion patterns that are used to generate synthetic video. Section 3 discusses the measurements of evaluating accuracy of sprite. Section 4 describes experiments based on the camera motion patterns. The last section concludes our paper.

II. COMPLEX CAMERA MOTION PATTERNS

In this paper, we present three camera motion patterns for synthetic video generation: Rotation, Affine, and Pan-Tilt-Zoom (PTZ). Each pattern generates frames such that the consecutive frames have overlapping areas with the previous image in the sequence.

In compared with the translational patterns we proposed in our previous work [6], these camera motion patterns cover specific area of original image rather than the whole original image. We also present two combined patterns which cover the whole images. We briefly explain these patterns in this section.

2.1 Rotation

The *rotation* pattern generates a sequences of sequential images based on spin around the center of the original image as in Fig. 1. The camera can spin 90 iterations

Fig. 1. Rotate Pattern

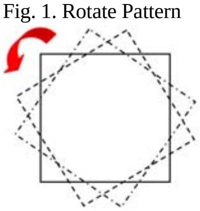
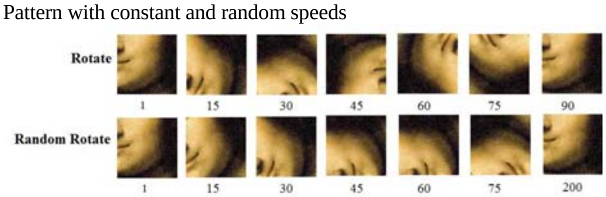


Fig. 2. Mona Lisa

with 4 degrees each time or spin 360 degrees with random speed less than 4 degrees. Fig. 3. shows sample images that are generated from Mona Lisa image (Fig. 2) using the rotation pattern with constant and random speeds.

Fig. 3. Sample images generated from Mona Lisa image using Rotate Pattern with constant and random speeds



2.2 Affine

The *affine* pattern generates a sequence of sequential images based on Fig. 4. According to Fig. 4, we move points A, B, and C less than 5 pixels both in x coordinate and y coordinate and then calculate the corresponding D points. We capture all the new points covered by the new quadrangle and generate new image by mapping the points on a rectangular grid. Fig. 6 present sample affine pattern images generated from Mississippi Delta image.

Fig. 4. Affine Pattern

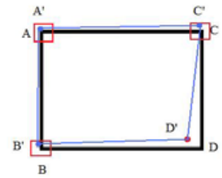


Fig. 5. Mississippi Delta (from NASA)

Fig.6 Sample Affine pattern images generated from Mississippi Delta image

2.3 Pan-Tilt-Zoom

The *Pan-Tilt-Zoom* (PTZ) pattern generates a sequence of sequential images based on Fig. 7. We firstly move the frame in horizontal direction (Pan), secondly move the frame in vertical direction (Tilt), and then zoom in or zoom out. Fig. 8. are sample PTZ pattern images generated from Mississippi Delta image.

Fig. 7. PTZ Pattern

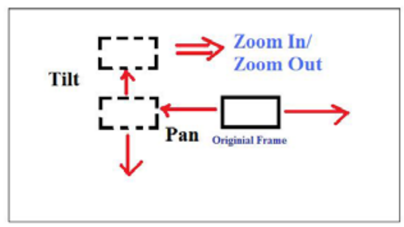


Fig.8 Sample PTZ pattern images generated from Mississippi Delta image

2.4 Combined Patterns

The Combined patterns include two modes.

Fig.9. CB1 pattern

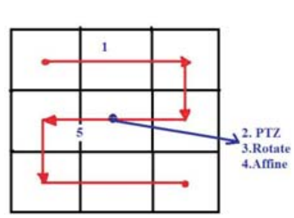
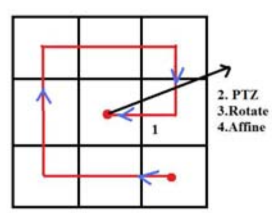


Fig.10. CB2 pattern



The first pattern named *Combined pattern I (CPI)* as in Fig. 9 follows this sequence: 1) Zigzag pattern at constant speed [6] 2) PTZ pattern, 3) rotation pattern, and 4) affine pattern.

The second one named *Combined pattern II (CPII)* as in Fig. 9 follows sequence: 1) Spiral pattern at constant speed [6] 2) PTZ pattern, 3) rotation pattern, and 4) affine pattern.

Both patterns cover whole area of original image.

III. MEASEUREMENT OF ACCURACY

In objective evaluation, PSNR is retrieved by comparing original frames and regenerated frames from the sprite:

$$PSNR = 10\log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20\log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right)$$

where MAX_I denotes the maximum error and MSE represents mean squared error.

The traditional sprite generation algorithms usually use *framePSNR [6]* . We used *picturePSNR [6]*, size, and error on motion parameters [6] in the translational and zoom patterns in our previous work. The *framePSNR* is computed between the original frame and generated frame for each frame in the sequence whereas the *picturePSNR* is computed between the original ground-truth image and the generated sprite.

Since the coverage of rotation, affine and PTZ patterns are not the whole image, the computation of the PSNR value between the original image and the generated sprite does not make much sense. So, we generate the ground truth image as the synthetic video is generated to check the accuracy. For translational patterns such as Zigzag and Spiral and for Zoom pattern alone, the ground truth image is the whole original image. For complex pattern like affine, PTZ, rotation, the ground truth image is part of the original image. For combined patterns, the ground truth image is the whole original image.

Pattern algebra files that includes motion parameters is also used to check the correctness of motion estimation for each frame in the sequence.

IV. EXPERIMENTS AND EVALUATION OF SPRITE GENERATION WITH RESPECT TO CAMERA MOTION PATTERNS

In this section, we show the results of applying these camera motion patterns. We show the ground truth images used by the complex camera motion patterns and also generated sprites. At the same time, we compare the difference between the ground truth and sprite generated.

In addition, we also show the global motion file that is used for tracking whether every prediction step of sprite generation works correctly.

4.1 Rotation

Fig. 11 presents the ground truth used and sprites generated on synthetic video with rotation pattern with constant and random speeds. By subjective evaluation, we can find the sprite generated is vaguer than the ground truth image. In other words, some details are missed.

Fig. 11. Ground truth and sprite images on Mississippi Delta image

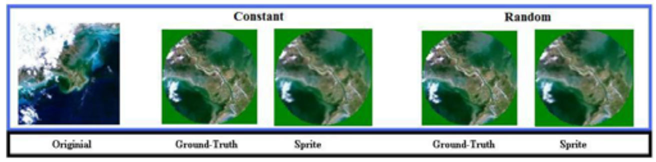


Table 1 presents the size difference between ground truth image and sprite. It shows the sprite generation works well on size but it also reveals errors on sprite generation.

Table 1. Size comparison for rotation pattern

	Original Mask	Sprite
Size	141*140	151*143

Figures 12 and 13 present the frame PSNR, average PSNR file and Picture PSNR for Mississippi Delta image for rotation and random rotation patterns, respectively. The trend of prediction is very similar except that there is more oscillation in random rotation pattern.

Fig. 12. PSNR comparison table for Mississippi Delta using Rotate pattern

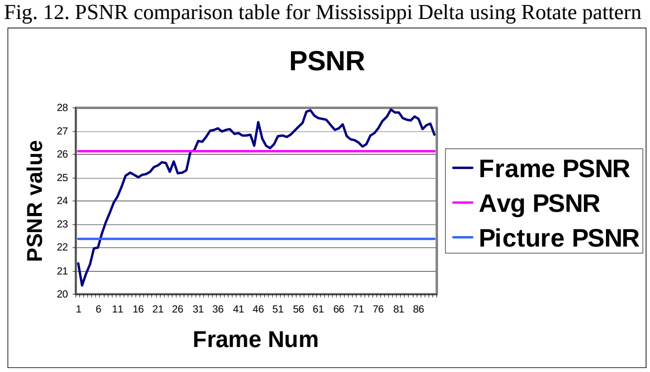


Fig. 13. PSNR comparison table for Mississippi Delta using RR pattern

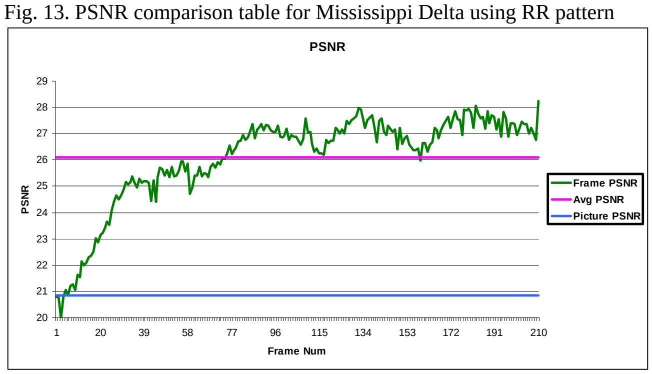
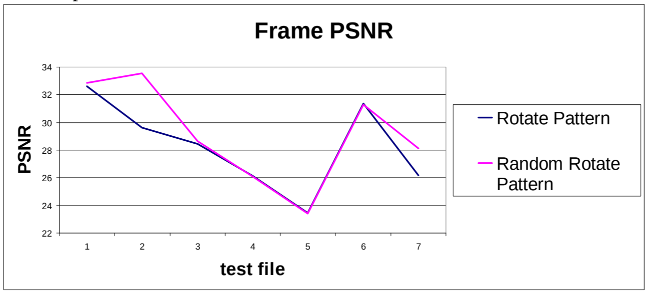


Fig. 14 compares the average of Frame PSNR of rotation and random rotation patterns of 7 test data. The prediction of rotation pattern is not very different between constant speed and variable speed. For some test data, the prediction of global motion parameters is almost the same for random and constant speeds.

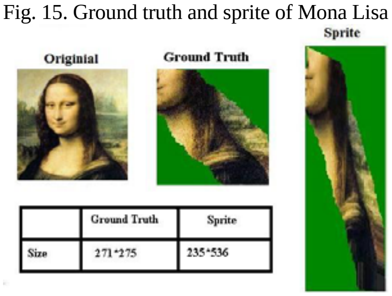
Fig. 14. Average Frame PSNR value of 7 test data on rotation and random rotation patterns



4.2 *Affine*

Fig. 15 shows the ground truth and sprites generated on synthetic video with affine pattern. The sprite generated by affine pattern reflects that the sprite generation should have small error boundary for sprite generation on affine pattern.

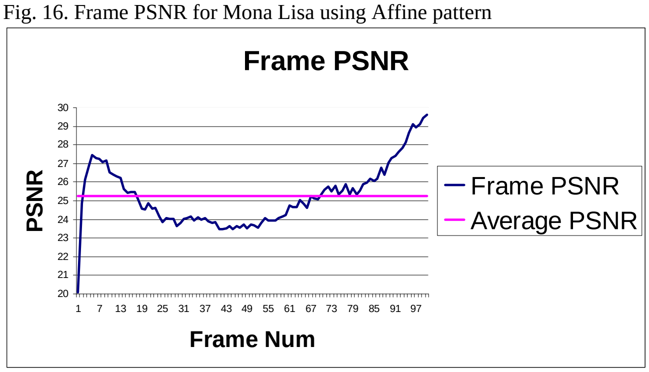
Fig. 15. Ground truth and sprite of Mona Lisa image and size difference



	Ground Truth	Sprite
Size	271*275	235*536

Since the size changes a lot, calculation of picture PSNR is not necessary. Figure 16 presents that the prediction of the sprite is initially wrong. Since in the affine motion of sprite generation, the similar area will be aligned and it does not yield good PSNR.

Fig. 16. Frame PSNR for Mona Lisa using Affine pattern

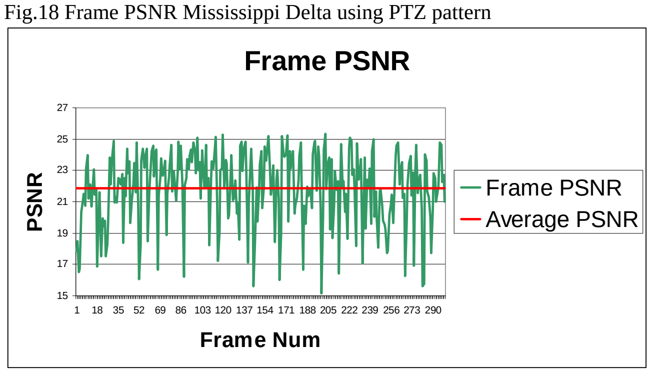


4.3 *Pan-Tilt-Zoom(PTZ)*

Fig. 17 indicates that the details of original ground truth are not clear and the shape of the prediction is not clear. According to Fig. 18, the sprite generated is not accurate in both size and clarity although the average PSNR value is acceptable.

Fig.17 Ground Truth and sprite of Mississippi Delta image on PTZ pattern

Fig.18 Frame PSNR Mississippi Delta using PTZ pattern



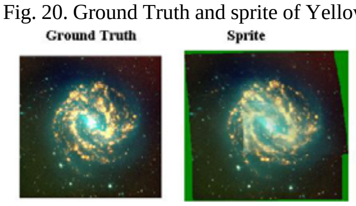
4.4 *Combined Pattern*

Figures 19 and 20 indicate that the sprite generation does not perform well on complex patterns which are composed of several basic camera motion pattern.

Fig. 19. Ground Truth and sprite of the Yellow Star image on CP1 pattern

	Ground Truth	Sprite
Size	450*450	826*614

Fig. 20. Ground Truth and sprite of Yellow Star image on CP2 pattern



	Ground Truth	Sprite
Size	450*450	518*475

Fig. 21. Frame PSNR of Yellow Star using CP1

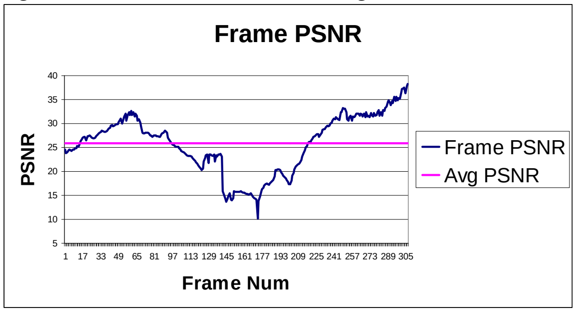
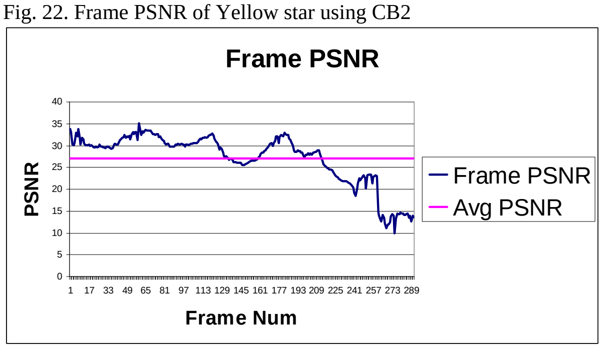


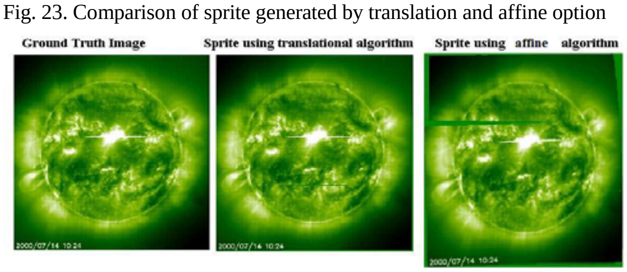
Fig. 22. Frame PSNR of Yellow star using CB2



In summary, CP1 and CP2 include both simple patterns and complex patterns. The frame PSNR value is relatively low when switching patterns. The sprite generation with enforced affine motion parameters is not very sensitive to the complex patterns. In Fig. 21, frames 121 through 171 are the period of complex pattern including affine, rotation and PTZ. In Fig. 22, frames 240 through 290 are the period containing affine, rotation and PTZ. All these information can be traced back from pattern algebra file.

Analyzing the middle part of the sequence for CP1 and CP2, we found that the enforcing affine motion estimation in sprite generation was not efficient in generating the translational patterns. Fig. 23 presents the sprite that is generated with the spiral pattern using enforced affine motion.

Fig. 23. Comparison of sprite generated by translation and affine option

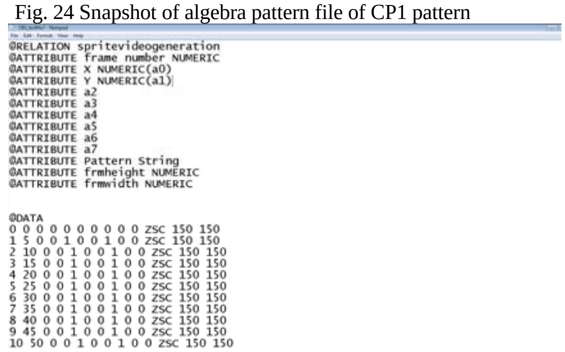


4.5 Pattern Algebra File (Ground Truth File)

We also generated ground truth file as in Fig. 24 to check the process of global motion estimation. This file can be used to regenerate pattern images and it can also be used to track the change of global motion parameters of the camera pattern. For example, we can find the corresponding pattern when the *framePSNR* value is very low and find the problem of sprite generation.

Fig. 25 presents sample results of our experiments.

Fig. 24 Snapshot of algebra pattern file of CP1 pattern



```
@RELATION spritevideogeneration
@ATTRIBUTE frame number NUMERIC
@ATTRIBUTE X NUMERIC(a0)
@ATTRIBUTE Y NUMERIC(a1)
@ATTRIBUTE a2
@ATTRIBUTE a3
@ATTRIBUTE a4
@ATTRIBUTE a5
@ATTRIBUTE a6
@ATTRIBUTE a7
@ATTRIBUTE Pattern String
@ATTRIBUTE frmheight NUMERIC
@ATTRIBUTE frmwidth NUMERIC


@DATA
0 0 0 0 0 0 0 0 0 0 ZSC 150 150
1 5 0 0 1 0 0 1 0 0 ZSC 150 150
2 10 0 0 1 0 0 1 0 0 ZSC 150 150
3 15 0 0 1 0 0 1 0 0 ZSC 150 150
4 20 0 0 1 0 0 1 0 0 ZSC 150 150
5 25 0 0 1 0 0 1 0 0 ZSC 150 150
6 30 0 0 1 0 0 1 0 0 ZSC 150 150
7 35 0 0 1 0 0 1 0 0 ZSC 150 150
8 40 0 0 1 0 0 1 0 0 ZSC 150 150
9 45 0 0 1 0 0 1 0 0 ZSC 150 150
10 50 0 0 1 0 0 1 0 0 ZSC 150 150
```

V. Conclusion and Future Work

In this paper, we proposed several new camera motion patterns to generate synthetic video for sprite generation. And we use PSNR, size, ground truth image, and pattern algebra global motion file to evaluate the quality of sprite in addition to frame PSNR. Our metrics with camera motion parameters and ground-truth images are better than frame PSNR to check the quality of the sprite. Our methods are also a better indicator of the weaknesses of the sprite generation method. As future work, we plan to implement an algebra tool to create the synthetic video generation according to pattern algebra and track the performance of sprite generation along with the change of camera motion patterns. We plan to improve the performance of our sprite generation algorithms.

References

[1] http://www.chiariglione.org/mpeg/standards/MPEG-

[2] T. Wiegand, G.J. Sullivan, G. Bjntegaard and A.Luthra, “Overview of the H.264/AVC video coding standard,” IEEE Transactions on Circuits and Systems for Video Technology vol. 13, pp. 560–576, July 2003.

[3] M. Kunter, P.Krey, A. Krutz and T. Sikora, “Extending H.264/AVC with a background sprite prediction mode,” Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on 12-15 Oct. 2008 Page(s):2128 – 2131

[4] J. Black, T. Ellis, and P. Rosin. “ A Novel Method for Video Tracking Performance,” in Proc. IEEE PETS Workshop, Oct.2003.

[5] A. Smolic, T. Sikora, and J.-R. Ohm, “Long-term global motion estimation and its application for sprite coding, content description and segmentation,” IEEE Trans. Circuits Syst. Video Technol., vol.9,pp. 1227-1242, Dec. 1999

[6] Yi Chen and RamazanS. Aygun. “Synthetic Video Generation to check the correctness of sprite generation,” *The First International Conference on Advances in Multimedia MMEDIA 2009 July 20-25, 2009 - Colmar, France*.

[7] Ç. E. Erdem and B. Sankur, "Performance Evaluation Metrics for Object-Based Video Segmentation," X European Signal Proc. Conf (EUSIPCO), September 4- 8, Tampere, Finland, 2000.

[8] A. Ishikawa, Panahpour Tehrani, M., Naito, S., Sakazawa, S., and Koike, A. 2008. Free viewpoint video generation for walk-through experience using image-based rendering. In Proceeding of the 16th ACM Int. Conf. on Multimedia (Vancouver British Columbia, Canada, October 26 - 31, 2008). MM '08. ACM, New York, NY, 1007-1008

Fig. 25 a) Rotation: Ground Truth Image and Sprite, b) Random Rotation Ground Truth Image and Sprite, c) Affine Ground Truth Image and Sprite, and d) Ground Truth Image and Sprite of combined pattern