# Inter-frame Affinity Loss for Consistent Video Style Transfer

Brian Tsan
EECS 286 - Fall 2020
University of California, Merced
btsan@ucmerced.edu

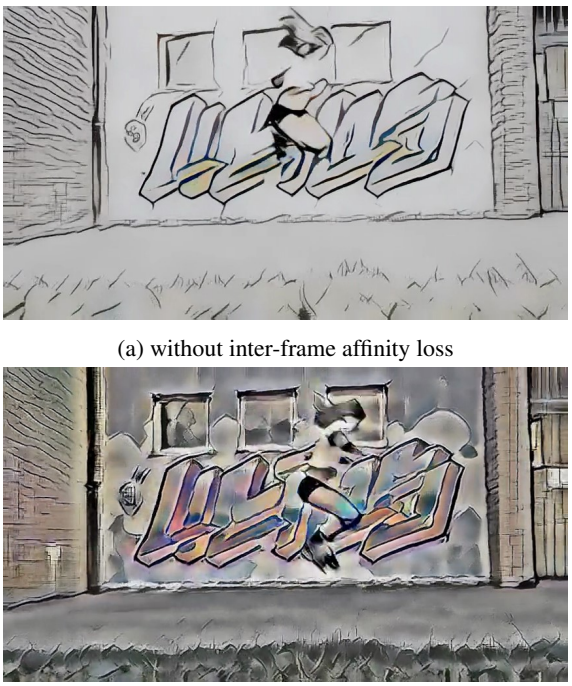
## Abstract

*Consistency is a current problem in applying style transfer to video. Although there has been recent advances in style transfer to still images, directly applying those methods to video results in undesirable flickering effects when each video frame is stylized independently. In parallel, recent advances in affinity-based visual correspondence learning can accurately approximate the transition between video frames as a linear transformation as an inter-frame affinity matrix. Based on those advances, this work proposes a novel inter-frame affinity loss to penalize deviations in the transition between video frames. Experimental results show that minimizing this inter-frame affinity loss produces more consistent videos and also reduces geometrical deformation from style transfer, allowing for a trade-off between temporal consistency and style transfer.*

## 1. Introduction

Neural style transfer methods have produced interesting results on still images. Gatys et al. [1] proposed extracting content and style features using pretrained VGG models. Style transfer can then be performed by iteratively optimizing image pixels to render content in arbitrary styles. As an iterative optimization process, this style transfer method is unsuitable for video applications, where many image frames need to be stylized. Johnson et al. [4] showed that the perceptual losses can also train a feed-forward neural network that can quickly perform style transfer in a single forward pass. Huang and Belongie. [3] not only proposes a network that can perform style transfer in real-time at 15 FPS, but also in arbitrary styles.

Directly applying these methods to videos would treat each frame as independent images. In this way, each stylized frame can be produced with noticeable variations from each other, even if frames are temporally contiguous and only slightly different from each other. This causes undesirable flickering artifacts when the stylized frames are played together, due to inconsistent style transfer and the geometric deformation of content. Inherently noisy image regions, such as the brick wall in Figure 1 is more susceptible to this undesirable effect.



(a) without inter-frame affinity loss

(b) with inter-frame affinity loss

Figure 1: The same network trained with inter-frame affinity loss preserves image color and intensity, while also maintaining more consistent style transfer in noisy image regions, such as the brick walls on the side of both images.

To address this video consistency issue, Ruder et al. [10] proposed using optical flow to define a temporal consistency loss. Although it produces qualitatively outstanding videos, their style transfer method is also a time-consuming iterative optimization process, which helped achieve their extremely consistent results. As such is the case, it would be very time consuming to run on long videos. Huang et al. [2] utilized the same temporal consistency loss to train a feed-

forward neural network to stylize videos while penalizing inconsistency, but the results are still noticeable less consistent than the iterative method, albeit significantly faster to produce.

In parallel with recent style transfer advances, visual correspondence learning based on inter-frame affinity have been proposed. Li and Liu et al. [5] used self-supervision to learn an accurate inter-frame affinity estimator that can then be applied to a wide range of visual correspondence tasks, such as video object tracking and semantic segmentation. The inter-frame affinity estimator is also quick to compute, requiring only a forward pass over each frame. The estimator produces an inter-frame affinity matrix, which represents the frame transition as the cross-correlation between pixels in each frame.

This work utilizes the inter-frame affinity matrix to define an inter-frame affinity loss penalizing changes between the source video affinity and the stylized video affinity. Intuitively, a stylized pixel in frame $A$ should correspond to the same pixel location in frame $B$ as it did in the source video frames. Using the source video affinity as a guideline, such stylized frames would appear more consistent with the original. Figure 1 shows that the inter-frame affinity estimator preserves the color and intensity from the original video. It also maintains more consistent style transfer in noisy image regions.

## 2. Related Work

### 2.1. Image Style Transfer

Gatys et al. [1] defined image content and style based on the features of deep convolutional neural networks, specifically the VGG architecture. Generally, style information is best defined at the later layers of feed-forward convolutional neural networks, whereas content information is better defined at earlier layers. At later layers, finer detailed pixel information has been lost and content information is more abstract.

Image content is simply defined as the feature representations extract from a convolutional layer of the neural network. Content loss is then defined as the mean squared error between the generated image and a given content image. In Equation 1 (1), let $C$ be the content image and $X$ be the generated image. $C^l$ is then the feature representation produced by at a convolutional layer $l$.

$$\mathcal{L}_c = \frac{1}{NM}\sum_{i,j}(C^l_{ij} - X^l_{ij})^2 \tag{1}$$

Style is similarly defined, except by using the Gram matrix of the convolutional feature representations. Using the Gram matrix defines style as the cross correlation between convolutional features. These correlations are given by Gram matrix $G$ where $G^l_{i,j}$ represents the inner product between the $i$th and $j$th features of feature map $F$ at convolutional layer $l$.

$$G^l_{i,j} = \sum_k F^l_{ik}F^l_{jk} \tag{2}$$

$$\mathcal{L}_s = \frac{1}{NM}\sum_{i,j}(S^l_{ij} - X^l_{ij})^2 \tag{3}$$

Style loss is then defined as the mean squared error between the Gram matrices of the style image $S$ and the generated image $X$. Gatys et al. [1] recommend defining style using multiple convolutional layers up to the later layers to generate more visually appealing images. Earlier layers preserves more local detail and incorporating them produces smoother output.

Their method iteratively minimizes the content loss (1) and style loss (3) with respect to pixels in an image. As an iterative method, it is unsuitable for stylizing videos with many image frames.

Johnson et al. [4] demonstrates that the perceptual losses (1)(3) defined by Gatys et al. [1] can be used to train a generative feed-forward neural network to more quickly stylize an input image. Once trained, the generative network can stylize an input content image in a single forward pass. Additionally, perceptual losses are demonstrated to generalize towards other image processing tasks, such as image super resolution. A shortcoming of this method is that a generative network must be trained per style.

$$\text{AdaIN}(x, y) = \sigma(y)\frac{x - \mu(y)}{\sigma(x)} + \mu(y) \tag{4}$$

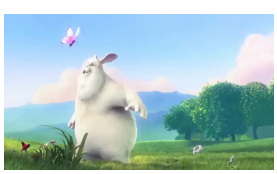
The AdaIN-Style [3] method supports arbitrary styles by training a generative network to decode feature maps extracted from content images with a pretrained VGG model. Those feature maps are normalized (4) to have a similar mean $\mu$ and standard deviation $\sigma$ as any arbitrary style image features. The decoder is trained while minimizing the content loss (1) and a style loss based on the mean and standard deviation of the activations from VGG layers. A forward pass of the trained model can quickly process and style an image.

Directly applying image style transfer methods to video frames independently of each other, likely causing each stylized frame to vary significantly. This may cause flickering, which is especially noticeable when even contiguous frames differ greatly. This is usually the case when styles are *noisy*, as in Figure 2. Inconsistencies can even be produced in still scenes, since slight pixel variations affect the generated output.

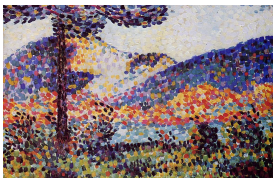
### 2.2. Video Style Transfer

Ruder et al. [10] utilizes optical flow to define a temporal loss for penalizing inconsistency in stylized videos. Optical

(a) Stylized



(b) Original



(c) Style

Figure 2: Independent style transfer may cause high variation among each video frames, especially with a noisy texture such as 2c

flow is also used to initialize the input image of the next frame by warping the previous styled frame. This guides the stylization process towards generating a consistent next frame.

$$\mathcal{L}_{temporal}(x, \omega) = \frac{1}{NM} \sum_{ij} (x_{ij} - \omega_{i,j})^2 \tag{5}$$

The temporal loss (5) penalizes any deviation in the generated frame image $x$ and the warped prediction $\omega$, which is the previous frame of $x$ warped by its optical flow. By stylizing the previous warped frame, any undesirable artifacts from the warping process is removed, while the temporal loss helps to avoid deviating from the warped prediction.

This method applies consistent style transfer to videos, but involves time-consuming iterative optimization as well as the added expense of calculating each frame's optical flow. Later methods [2] utilize the temporal loss to train fast generative networks..

### 2.3. Temporal Correspondence

Parallel to advances in style transfer, Li and Liu et al. [5] used self-supervised learning to train an inter-frame affinity estimator for performing visual correspondence tasks. The inter-frame affinity matrix $A$ represents the linear transformation between frames, where $A_{i,j}$ is the similarity of pixel $i$ in one frame to pixel $j$ in the other. Intuitively, if $A_{i,j} = 1$ then pixel $i$ is a copy of pixel $j$.

$$A_{i,j} = \frac{e^{x_i^\intercal y_j}}{\sum_k e^{x_k^\intercal y_j}} \tag{6}$$

Inter-frame affinity matrix $A$(6) is calculated by the dot product between feature maps $x$ and $y$, then applying softmax to each column to sharpen the estimation. The feature maps $x$ and $y$ are extracted from images by a convolutional neural network [5]. This networks is train completely self-supervised on videos and the resulting affinity estimator is shown to be useful on a variety of visual tasks, such as object tracking and image segmentation.

## 3. Losses

This paper utilizes the inter-frame affinity estimator [5] to develop a loss for improving the consistency of stylizing videos with a generative network. This paper uses the AdaIN-Style [3] model since it is fast and supports arbitrary styles.

### 3.1. Perceptual Losses

Huang and Belongie [3] trained their model using the same perceptual content loss (1) defined in previous works [4] but chose not to use the Gram matrix style loss (3). Instead, they use a style loss (7) that penalizes the error between the means and standard deviations between the generated image features $X^l$ and the style image features $S^l$ where $l$ denotes the corresponding network layer.

$$\mathcal{L}_s = \left\|\mu(S^l) - \mu(X^l)\right\|_2 + \left\|\sigma(S^l) - \sigma(X^l)\right\|_2 \tag{7}$$

This style loss has also been explored by Li et al. [6] and matches the adaptive instance normalization (4) in its concept of aligning the mean and standard deviation of content image features with the style image features.

### 3.2. Inter-frame Affinity Loss

In addition to the perceptual losses, this paper proposes a novel affinity-based loss (8) that promotes the temporal consistency of stylizing video frames. Like optical flow, the inter-frame affinity matrix also describes the transition between temporal frames. However, unlike an optical flow map, the inter-frame affinity matrix represents a linear transformation and so is unable to be used to detect disocclusion [10]. Even so, it is fast to accurately estimate the affinity between images [5].

$$\mathcal{L}_a = \frac{1}{NM} \sum_{i,j} (\mathcal{G} \odot ((A + I\epsilon) - \hat{A})^2)_{ij} \tag{8}$$

Besides taking the mean squared error between the affinity of generated frames $\hat{A}$ and the target affinity of the original content frames $A$, a diagonal matrix is added to the

(a) Original frames

(b) without inter-frame affinity loss

(c) with inter-frame affinity loss

Figure 3: Inter-frame affinity loss preserves more color and objects are more distinct.

target affinity to perturb its main diagonal by some small value $\epsilon$. This is to help the generative network learn to generate frames that are more similar to their previous frames, countering any potentially faulty training signal from fast moving and chaotic scenes.

The fast panning scene in Figure 3 causes blurring artifacts to appear during style transfer as objects enter and exit the scene. Weighting down the inter-frame affinity loss using a gaussian filter $\mathcal{G}$ resolves this issue. This problem belongs to the inter-frame affinity loss assuming the contents of a scene is static. This same blurring does not occur when the network is not trained with inter-frame affinity loss.

### 3.3. Temporal Loss

Besides the inter-frame affinity loss, the mean squared error between each pair of adjacent frames is used as an additional loss $L_{temp}$. This, along with the diagonal perturbation $\epsilon$ in (8), are designed to improve stability among stylized frames by penalizing significant differences between frames.

$$\mathcal{L} = \lambda_1 \mathcal{L}_c + \lambda_2 \mathcal{L}_s + \lambda_3 \mathcal{L}_a + \lambda_4 \mathcal{L}_{temp} \tag{9}$$

The overall loss function (9) is the sum of the 2 perceptual losses, the inter-frame affinity loss, and the temporal loss, each weighted by some values $\lambda_i$.

## 4. Experimental Setup

To compute the inter-frame affinity loss (8) the affinity estimator provided by Li and Liu et al. [5] is used without any further training. The style transfer network architecture follows the implementation of the orignal AdaIN-Style [3] and uses VGG19 layers up to `relu_4_1` as an encoder.

### 4.1. Training

The model is initially trained with perceptual losses to stylize still images using the MS-COCO image dataset [7] and the WikiArt [9] dataset for content and style, respectively. Then the DAVIS 2017 unsupervised training set is used for training the on videos with the full loss function (9). The dataset consists of 90 short videos with a total of over 5000 frames. Frames are loaded in batches where each frame is up to 5 frames away from the next, to prevent having overly similar frames in the same batch, such that their affinities are approximately the identity matrix.

A shortcoming of using the inter-frame affinity is that it assumes the objects within a pair of frames to be the same as each other. In order for the inter-frame affinity to be accurately estimated, the interval between frames in a batch must not be too far so that objects suddenly disappears or jumps significantly in-between frames. This was also a problem in previous work on affinity-based visual correspondence learning [12] but intervals of up to 10 appears to be unproblematic for this dataset.

Additionally, the formulation of the inter-frame affinity loss (8) addresses the issue of objects dynamically entering and exiting the scene, by weighting the loss to be focused on the center of the frame scene. The values of weights $\lambda_i$ in (9) are $1, 3, 10000$, and 10 for $\mathcal{L}_c, \mathcal{L}_s, \mathcal{L}_a$, and $\mathcal{L}_{temp}$, respectively. A value of $0.1$ is used for $\epsilon$ in the inter-frame affinity loss (8).

## 5. Results

The qualitative results of training with the inter-frame affinity loss (8) are shown in Figure 1 in comparison to a baseline model trained without the inter-frame affinity loss (8) and temporal loss (9). It should be noted that the results produced by the baseline model still strongly preserves the original content, such that stylized videos are already very visually consistent. The effect of training with the inter-frame affinity loss (8) is best observed across multiple different styles.

The network trained with the inter-frame affinity loss preserves the color and intensity of the content pixels, even when the network trained without inter-frame affinity would otherwise render a black and white scene. Although the resulting frames may also have less noticeable flickering, due to being densely colored, and a black and white scene can still be produced by adjusting the color saturation in post-processing, this effect may defeat the purpose of style transfer when the intention is to heavily stylize a video, such as rendering something in black and white lines or strokes.

Figure 4 shows that inter-frame affinity loss results in more strongly pronounced objects in the background, as opposed to Figure 4a which obscures the background to suit the style image. This indicates that inter-frame affinity loss penalizes loss of detail from the content image, even if discarding some would better suit the style.

### 5.1. Color Preservation

To preserve the original content affinity in between the stylized frames, the network has learned to preserve the intensity of the pixels and also their color, to some degree. However, the effect is different from preserving the content using a stylization strength $\alpha$ parameter to adjust the degree of stylization.

$$S' = (1 - \alpha)C + \alpha \text{AdaIN}(C, S) \tag{10}$$

Although using lower values of $\alpha$ in (10) can preserve the content $C$ and its color, it also reduces the degree of style $S$ that is transferred. Using affinity not only preserves the color and intensity of the content, but also maintains the degree of style transfer. By preserving just the color and intensity of the content, the affinity estimator can avoid being confused by different objects having similarly colored pixels.

Figure 5 shows that the network trained with inter-frame affinity loss not only preserves color, but also maintains a stronger degree of style transfer, compared to a network trained without inter-frame affinity loss. Although color is preserved by using a lower style transfer strength $\alpha$, the degree of style transfer is noticeably reduced. It should also be noted that this effect is not necessarily desirable in style transfer, if the intention is to transfer not just texture, but also color from the style image.

(a) without inter-frame affinity loss

(b) with inter-frame affinity loss

(c) content

(d) style

Figure 4: Strong color preservation is not necessarily desirable and may obscure other aspects of a style. Although texture is transferred to a similar degree in both images, 4b is less suitable for the *emptiness* of the style image.

### 5.2. Geometric Consistency

For some styles, it is noticeable that the style transfer seems more subtle, especially when the styles involve finer-grained texture. Although the style transfer network trained with inter-frame affinity achieves a significant degree of style transfer with coarse styles, such as rendering the outlines of objects in different strokes, it also avoids adding excess shapes where there originally was nothing. Adding new shapes and causing geometric deformation to an image may result in a less accurate inter-frame affinity matrix, so a side-effect of the inter-frame affinity loss is *smoother* stylization that strongly conforms to the original geometry of

(a) original

(b) without inter-frame affinity loss $\alpha = 0.3$

(c) without inter-frame affinity loss $\alpha = 0.6$

(d) without inter-frame affinity loss $\alpha = 1.0$

(e) with inter-frame affinity loss. $\alpha = 1.0$

Figure 5: Color is maintained without reducing the style transfer strength $\alpha$.

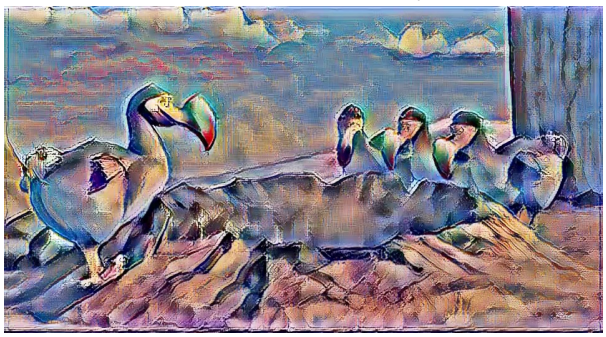
the content. This effect is similar to using the total variation loss proposed by Mahendran and Vedaldi [8] that is commonly used in style transfer to reduce the formation of style transfer artifacts, such as the excessive formation of shapes where there were none.

$$\mathcal{L}_v = \sum_{ij} \sqrt{(x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2} \quad (11)$$

This suppression of noisy artifacts appearing notably results in more consistent videos, for which higher frequencies of artifacts appearing would otherwise cause more noticeable flickering. This smoothing effect may not be completely desirable in style transfer, since some styles that focus on geometric deformation or complex shapes may not be transferred well. Maintaining a consistent geometric deformation throughout a video is a challenging problem which the method by Ruder et al. [10] handles well, but is an iterative optimization method that is time consuming for lengthy videos.

In Figure 6 the complex texture of the style image is less noticeable in the image generated by the network trained with inter-frame affinity loss. Since color is also preserved, the overall effect of style transfer is much less noticeable, indicating that such styles are less effective with this method. That style image incorporates smaller shapes into overall patterns. While the grainy texture is adopted, the more complicated curves are not apparent in the generated

(a) without inter-frame affinity loss



(b) with inter-frame affinity loss

(c) content

(d) style

Figure 6: Original texture is more strongly preserved, suppressing formation of complex textures from transferred style. Without inter-frame affinity loss, the background texture is more strongly deformed.

image.

### 5.3. Style Interpolation

Any number of arbitrary styles can be interpolated in AdaIN-Style. For a set of $K$ styles with respective weights $w_0, w_1, w_2, ..., w_K$, simply take the weighted average (12) of their encoded features.

$$C' = \frac{\sum_k w_k \text{AdaIN}(C, S_k)}{\sum_k w_k} \tag{12}$$

After interpolation, the feature map $C'$ can be decoded by the generative network to generate an image using mutliple styles. For videos, this mechanism can be used to gradually transition styles over multiple frames. Since inter-frame affinity loss preserves color and overall texture, this transition appears more subtle in Figure 7.

## 6. Conclusion

Using the inter-frame affinity loss to train style transfer models improves the appearance of videos by preserving the original color and shape. It also reduces the frequency of style transfer artifacts, resulting in less inconsistencies between video frames. The preservation of color is not necessarily desirable in style transfer, such as when the result is intended to be discolored. Contemporary work [11] on video style transfer primarily focuses on using optical flow to correct the consistency between frames. In future work, utilizing both optical flow and inter-frame affinity may further enhance temporal consistency in video style transfer. The current inter-frame affinity loss (8) does not fully address videos where objects dynamically enter and exit the scene, so optical flow may possibly be used to check the accuracy of the inter-frame affinity matrix and appropriately weight the loss.

## References


[1] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style. *CoRR*, abs/1508.06576, 2015. 1, 2

[2] H. Huang, H. Wang, W. Luo, L. Ma, W. Jiang, X. Zhu, Z. Li, and W. Liu. Real-time neural style transfer for videos. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7044–7052, 2017. 1, 3

[3] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization, 2017. 1, 2, 3, 4

[4] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, 2016. 1, 2, 3

[5] X. Li, S. Liu, S. D. Mello, X. Wang, J. Kautz, and M.-H. Yang. Joint-task self-supervised learning for temporal correspondence. In *NeurIPS*, 2019. 2, 3, 4

[6] Y. Li, N. Wang, J. Liu, and X. Hou. Demystifying neural style transfer. *CoRR*, abs/1701.01036, 2017. 3

[7] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár. Microsoft coco: Common objects in context, 2015. 4

[8] A. Mahendran and A. Vedaldi. Understanding deep image representations by inverting them, 2014. 6

[9] K. Nichol. Painter by numbers. https://www.kaggle.com/c/painter-by-numbers. Accessed: 2020. 4

[10] M. Ruder, A. Dosovitskiy, and T. Brox. Artistic style transfer for videos. *CoRR*, abs/1604.08610, 2016. 1, 2, 3, 6

[11] W. Wang, J. Xu, L. Zhang, Y. Wang, and J. Liu. Consistent video style transfer via compound regularization. In *AAAI Conference on Artificial Intelligence*, February 2020. 7

[12] X. Wang, A. Jabri, and A. A. Efros. Learning correspondence from the cycle-consistency of time. *CoRR*, abs/1903.07593, 2019. 4

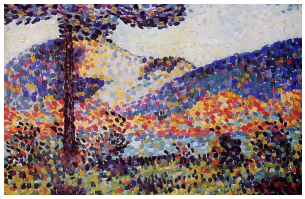
(a) Original frames

(b) Without inter-frame affinity loss. Uses Styles 1 and 2.

(c) With inter-frame affinity loss. Uses Styles 1 and 2.

(d) Style 1

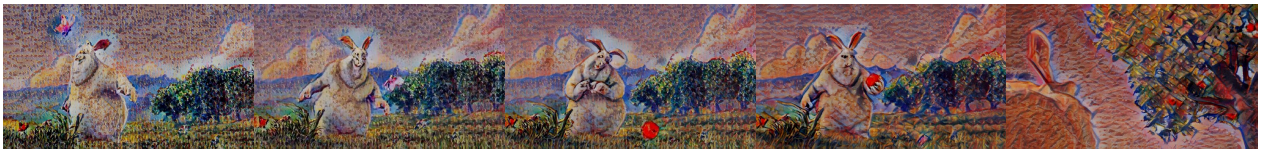
(e) Style 2



(f) Style 3

(g) Style 4

(h) Original frames



(i) Without inter-frame affinity loss. Uses Styles 3 and 4.

(j) With inter-frame affinity loss. Uses Styles 3 and 4

Figure 7: Gradual style interpolation over multiple frames. The effect of interpolation is less pronounced in the video produced with inter-frame affinity loss, since original color is preserved in 7c and texture in 7j.