Animation Video Resequencing with a Convolutional AutoEncoder

Shang-Wei Zhang[1], Charles C.Morace[1], Thi Ngoc Hanh Le[1], Chih-Kuo Yeh[1], Shih-Syun Lin[2], Sheng-Yi Yao[1], and Tong-Yee Lee[1]

[1] Department of Computer Science and Information Engineering, National Cheng Kung University, Taiwan

[2] Department of Computer Science and Engineering, National Taiwan Ocean University, Taiwan

I. Abstract

We introduce a data-driven solution to problem of general 2D animation resequencing. Given an unordered collection of images, the proposed method can create new "as-smooth-as-possible" animation of arbitrary length or select suitable in-between images for a set of key-frames. Our framework involves two phases. First, a denoising autoencoder is trained to extract a lower dimensional representation of an image so that the temporal coherence of images can be sufficiently measured. Then, the trained encoding network maps a new collection of images to their lower dimensional embedding, when we generate a variety of animations by traversing an approximated animation manifold. We describe the autoencoder's network architecture and training procedure in detail and give two path-finding algorithms, one for key-frame in-between selection and another for animation synthesis. In contrast to previous works, our proposed technique does not require fine-tuning of parameters and applies to a variety of image styles. Experimental evaluation proves our proposed method can generate appealing results.

II. Proposed Approach

Given an unordered collection of images as input, our proposed system can decide suitable in-between images for a set of key-frames, or synthesize new animation sequences which are locally "as smooth as possible". To control the output, a user has the option to supply either a set of key-frame images or a starting frame and sequence length; then the system will create an animation with the desired properties.

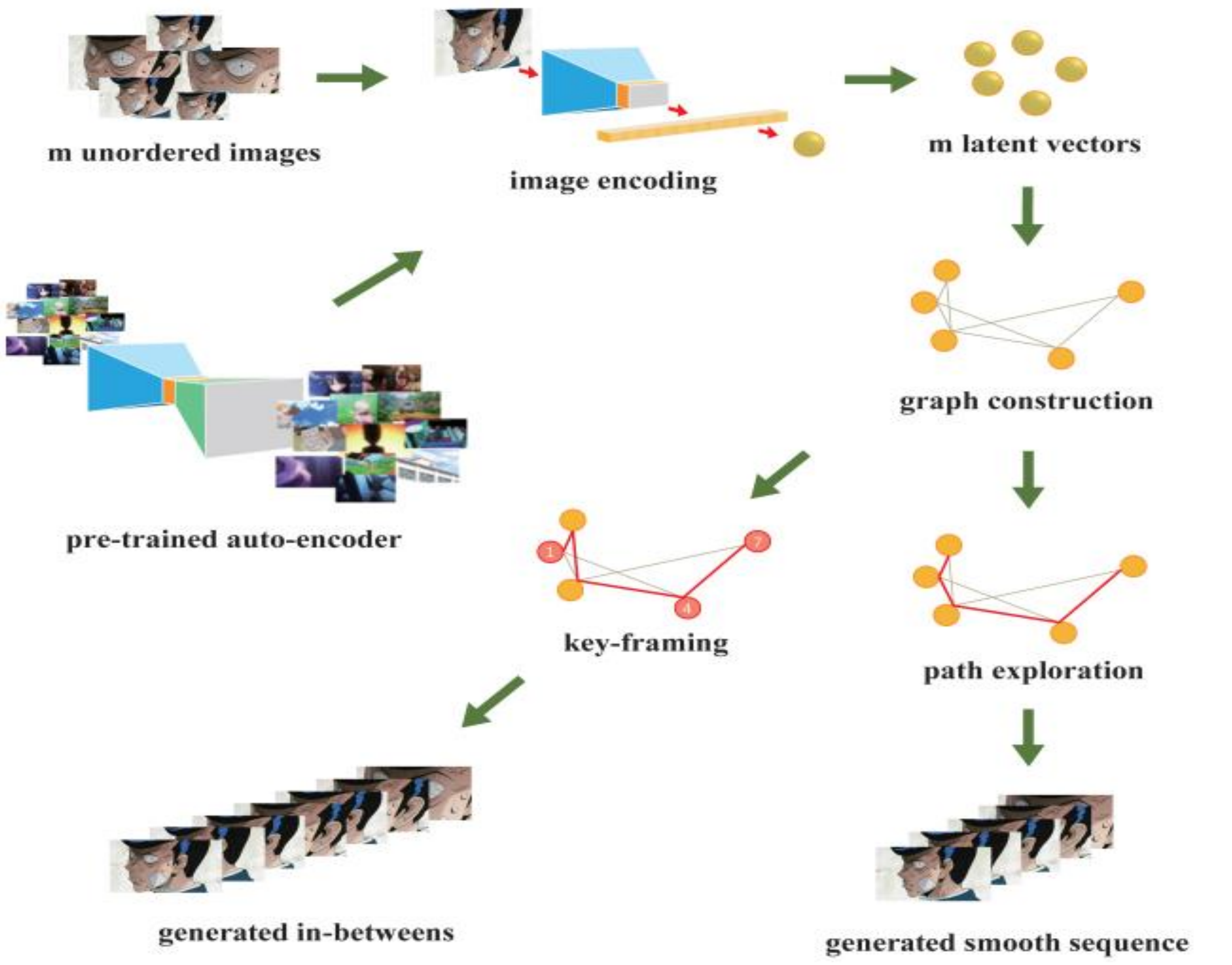

Figure 1: The system overview.

We firstly train a denoising autoencoder on a collection of animation images downloaded from the internet. After training the autoencoder, we apply the encoding network to a collection of images supplied by the user to obtain lower-dimensional latent vector representations. From these latent vectors, we compute a Euclidean minimal spanning tree (MST), and the proposed key-frame pathfinding uses the MST to generate in-between images. To synthesize new animations of arbitrary length, we compute a path-connected proximity graph and employ a Monte Carlo method to find a path that is as smooth as possible.

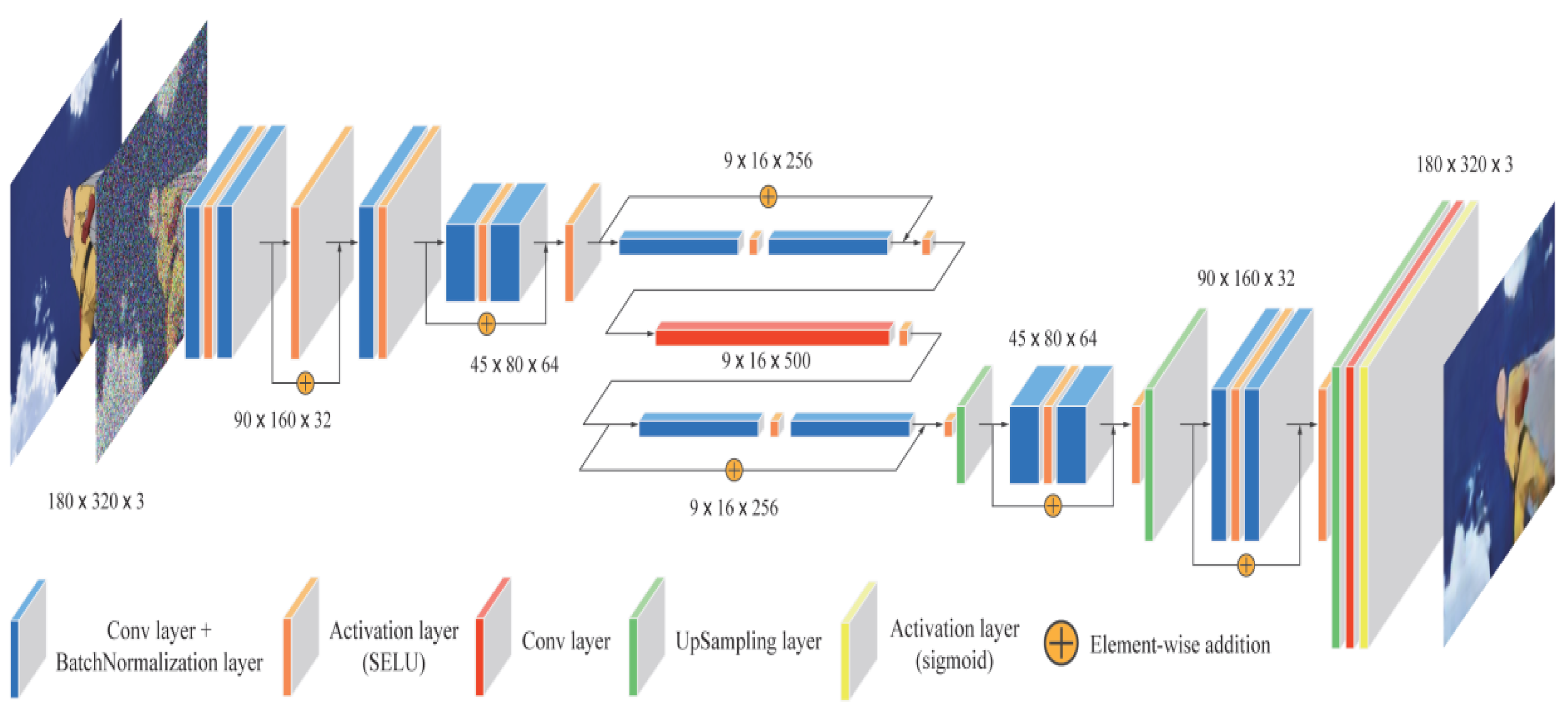

Figure 2: The network architecture of the proposed autoencoder.

Figure 3: Results were created by randomly selecting a contiguous group of eight frames from an animation generated with the proposed path exploration method.

In the phase of generating animation sequences, a key-framing method and a path exploration are proposed. In the key-frame pathfinding, the MST is employed to reduce in-betweens by traversing the path from one key-frame node to another. The paths connecting key-frame nodes in an MST are well suited for finding in between images. In the proposed path exploration method, we employ a Monte Carlo technique for synthesizing new animations with the desired length.

III. Results

To generate key-frame results, we examined the MST to guide key-frame selection and return precisely six in-between images. These results generated with the proposed method are shown in figure 3. In general, the user cannot directly control the number of in-between images returned for arbitrary key-frame selection. However, using the linear embedding of the MST for visualization provides a useful way to select key-frames that produce the desired number of in-betweens.

In the future, we would like to extend our work to a supervised learning framework which considers the temporal distance of the training data to solve above problems.

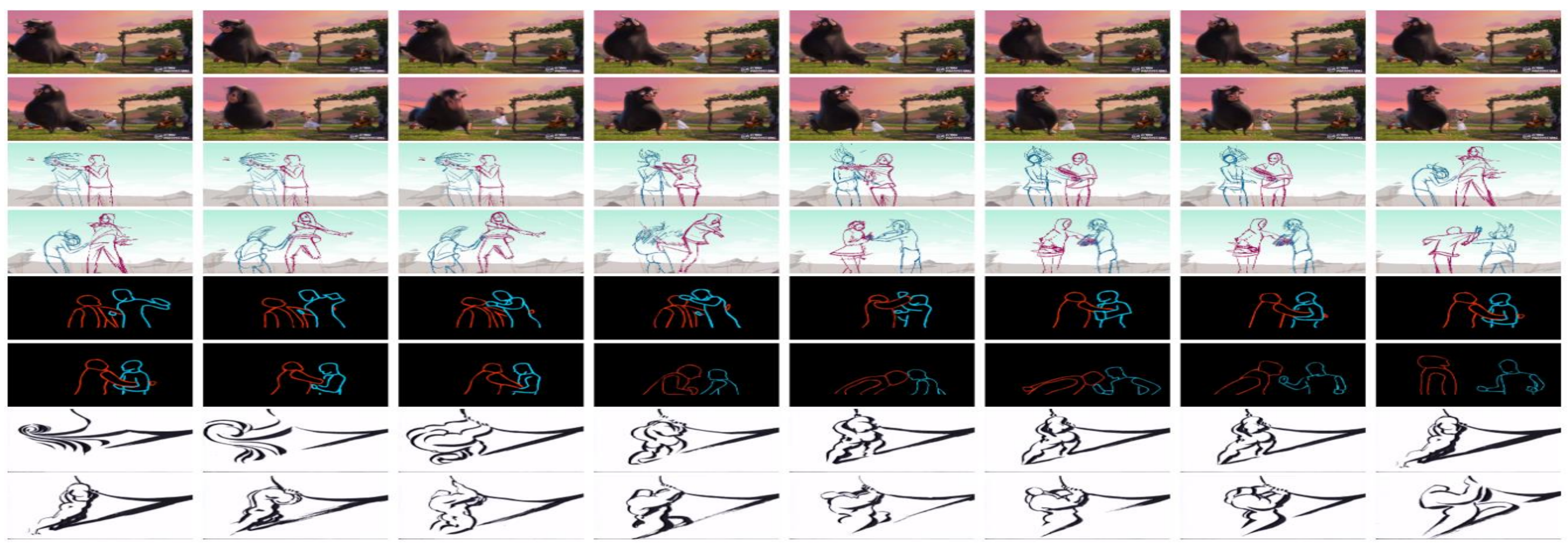

Figure 4: Results of our proposed key-frame method.

IV. Acknowledgment

We would like to thank the reviewers for many helpful comments. This work was supported in part by the Ministry of Science and Technology (107-2221-E-006 -196 -MY3, 108-2221-E-006 -038 -MY3, MOST-106-2221-E-019-069-MY2, and MOST-108-2221-E 019-038-MY2)