Gated Recurrent Networks for Video Super Resolution

Santiago López-Tapia ∗, Alice Lucas †, Rafael Molina ∗, Aggelos K. Katsaggelos †

∗ Depto. de Ciencias de la Computación e I.A., University of Granada, Granada, Spain
† Dept. of Electrical Engineering and Computer Science Northwestern University Evanston, IL, USA

Email: {sltapia, rms}@decsai.ugr.es, alicelucas2015@u.northwestern.edu, aggk@eecs.northwestern.edu

Abstract—Despite the success of Recurrent Neural Networks in tasks involving temporal video processing, few works in Video Super-Resolution (VSR) have employed them. In this work we propose a new Gated Recurrent Convolutional Neural Network for VSR adapting some of the key components of a Gated Recurrent Unit. Our model employs a deformable attention module to align the features calculated at the previous time step with the ones in the current step and then uses a gated operation to combine them. This allows our model to effectively re-use previously calculated features and exploit longer temporal relationships between frames without the need of explicit motion compensation. The experimental validation shows that our approach outperforms current VSR learning-based models in terms of perceptual quality and temporal consistency.

Index Terms—Video, Super-resolution, Convolutional Neural Networks, Recurrent Neural Networks

I. INTRODUCTION

Image Super-Resolution (SR) is one of the fundamental low-level vision problems. It consists of recovering a High-Resolution (HR) image from a given set of Low-Resolution (LR) images. In recent years, the introduction of high and ultra high definition displays has increased the demand for methods to convert already existing LR videos into HR ones. This is known as Video Super-Resolution (VSR), a special case of SR where the input and output are sequences of LR and HR video frames, respectively. The formation of each LR image from its corresponding HR image can be written as:

\[ y = \downarrow (x \otimes k) + \epsilon, \]

where \( x \) is the HR image, \( y \) is the LR one, \( x \otimes k \) represents the convolution of \( x \) with the blur kernel \( k \), \( \downarrow \) is a downsampling operator (typically bicubic downsampling) and \( \epsilon \) is the noise, usually Additive White Gaussian Noise (AWGN).

We can divide current (V)SR methods into two categories: model-based and learning-based. Model-based approaches explicitly define the LR image formation model (see Eq. 1) and typically recover the HR image by optimizing an energy function built from the LR image formation model [1], [2], [3]. In the case of learning-based algorithms, most of them do not explicitly define or use the image formation model, instead they use a large training databases of HR and LR image/sequence pairs to learn a mapping from the LR observations to the HR. When learning from data, Convolutional Neural Networks (CNN) have become a popular tool due to their high performance in other vision based tasks. In recent years, several VSR CNN-based models have been proposed: Caballero et al. [4] jointly train a spatial transformer network and an SR network to warp the video frames, the approach benefits from sub-pixel information. Tao et al. [5] propose to increase the performance of [4] by jointly upsampling and motion compensation (MC). Liu et al. [6] propose to train a neural network to learn the temporal dependency between input frames increasing the quality of the HR prediction. Kappeler et al. [7] propose to train a CNN which takes bicubically interpolated LR frames as input and learn the direct mapping that reconstructs the central HR frame. Following [7], Lucas et al. introduce in [8] a deep residual network trained using feature and adversarial losses that increased the perceptual quality of the output. Finally, in [9] we introduce a CNN GAN model that use the LR image formation model and a spatial-constraint to further increase the perceptual quality without the introduction of visual artifacts.

Despite all the work carried out in VSR in recent years using CNN-based models, few of them have used recurrent architectures. Recurrent Neural Networks (RNN) have been applied with great success to speech recognition and other tasks which require processing a time sequence, see [10], [11] for video related problems. One of the most frequently use recurrent units is the Gated Recurrent Unit (GRU) [12], which efficiently exploits the existing correlation within long sequences. However, in the case of VSR, these types of architectures have been hardly used and all the proposed methods employ simpler recurrent units (the result of processing the previous frame is used as input to the current one in contrast to general GRUs which use gated operations to update an internal state). In [13], the authors propose to use Recurrent Convolutional Layers to exploit the relations between frames more effectively. The network is very shallow since it does not use residual blocks.
More recently, a very deep residual was proposed in [14]. The models use a Convolutional Recurrent Neural Network (RCNN) that uses the super-resolved frame in the previous time step.

In this work, we propose a new RCNN that employs more effectively the previous time step information. Our model, inspired by the deformable convolutions introduced in [15], defines and utilizes fast deformable attention modules. These modules are used to align previously calculated features to the current time step ones. Then, our model combines them using a gated sum. Experiments show that this new architecture outperforms current VSR state-of-art methods in terms of PSNR and time consistency.

The rest of the paper is organized as follows: Section II-A presents our new recurrent model for VSR. In section III, we describe and discuss our experiments with the proposed model and comparison with state-of-art VSR algorithms. Finally, conclusions are presented in section IV.

II. Model description

In this work we use \( y_t \) to denote the LR frame at time \( t \) in a video sequence and \( x_t \) its corresponding HR. We model the process of obtaining a LR observation from a HR using Eq. 1. Following the literature [4], [5], [6], [7], [8], we assume that the noise is negligible \( (\epsilon = 0) \) and that the downsampling process \( (\downarrow (x \oplus k)) \) can be modeled using bicubic downsampling.

A. Architecture

The architecture of our proposed model, Gated Recurrent Video Super Resolution (GR-VSR), can be seen in Fig. 1. Our network is a recurrent network that uses the features \( h_{t-1} \) obtained at the previous time step from the convolution operation located right before the last upsampling module (they constitute the hidden state of our network) together with the features \( r_{t-2:t+2} \) calculated using \( y_{t-2:t+2} \). Using both, \( h_{t-1} \) and \( r_{t-2:t+2} \), and a deformable attention module (see Section II-B) we calculate and align with \( y_t \) the features \( \hat{h}_{t-1}^2 \) and \( \hat{h}_{t-2:t+2} \). These features are then concatenated and transformed using 8 residual blocks to produce \( \hat{h}_t \). The hidden state \( h_t \) is finally calculated as:

\[
\hat{h}_t = z_t \odot \hat{h}_t + (1 - z_t) \odot \hat{h}_{t-1}, \tag{2}
\]

where \( \odot \) indicates element-wise multiplication and \( z_t \) is a weight matrix with the same size as \( \hat{h}_{t-1} \) and \( \hat{h}_t \) which is calculated using a gate network. By using this update rule, our model is able to decide how much information from the current state will be added to the hidden state and how much of such information will be forgotten. Notice that this can be seen as a long skip connection. It is used in conjunction with residual blocks to construct deep CNNs [16]. It is important to mention that this connection is not used to increase the depth of the model (see [16]) but to allow the feedback provided by future frames to reach previous time steps during training. Furthermore, notice also that the use of this residual connection is not effective without alignment. Without alignment, the movement of the objects in the scene will cause the model to combine information from different locations in the scene, this will damage the HR prediction on those locations.

As it can be seen, the proposed architecture propagates the features contained in \( h_t \) instead of the predicted pixel values of \( \hat{x}_t \) as done in [14]. Our approximation allows to propagate more information through time, potentially encoding information from multiple time steps. For each spatial location our model encodes and transmits to the next step a vector of information, instead of just a value. This combined with the previous update rule (see Eq. 2) allows our model to propagate thought time much more information and during more time steps.

The use of a hidden state poses the problem that it has to be initialized. Though it can be initialized by a 0 vector, like in RNN used for text processing, we have detected in our experiments that this not only damages the prediction of the first couple of frames, but also, and more importantly, it makes the training very unstable. Thus, for the first frame prediction we propose the use of an auxiliary network \( g \phi \) identical to the proposed one but without the recurrent connection and related modules (deformable attention and gate network). Note that, although the use of this method increases the processing time, this is negligible for long video sequences since it is only needed for the first frame.

It is interesting to note that we can establish a parallelism between our proposed architecture and one of the GRU modifications: minimal GRU (see [17]). This minimal GRU performs the following operations:

\[
\hat{h}_t = \phi(W_h s_t + U_h h_{t-1} + b_h) \tag{3}
\]

\[
z_t = \sigma(W_z s_t + U_z h_{t-1} + b_z)
\]

\[
h_t = z_t \odot \hat{h}_t + (1 - z_t) \odot h_{t-1},
\]

where \( s_t \) is an input vector, \( W_\ast \) and \( U_\ast \) are learnable weight matrices, \( b_\ast \) biases, \( \sigma \) the sigmoid activation and \( \phi \) the tanH activation. As it can be seen, the first calculation depends on the input and the previous state. This operation corresponds in our model to the 8 residual blocks. Finally, the second and third calculation corresponds to our gated residual connection. Notice that the GRU it is not appropriated for image processing due to the lack of depth (no residual connections) and, in contrast to our model, it does not have a mechanism to align features.

B. Deformable attention

As we have already indicated in section II-A our model makes use of the deformable attention to align the features. Our deformable attention module is a modification of the deformable convolutions. Deformable convolutions were first proposed in [18] and enhanced in [15] to handle with more deformations. They correspond to the use of the following convolution operation

\[
f^l(m) = \sum_{\Delta m_{m,n}=1}^{N} w_n a_{m,n} f^{l-1}(m + n + \Delta m_{m,n}), \tag{4}
\]
where \( f^l(m) \) denotes the feature vector in layer \( l \) at location \( m \), \( n \) is a position of the convolutional kernel and \( \Delta m_{m,n} \) and \( a_{m,n} \) are learnable offsets and modulation scalar, respectively. These offsets and modulation parameters are calculated using another convolution layer for each \( m \) location in the image.

In our deformable attention module we impose to the model in Eq. 4 the following constraints: \( \sum_N a_{m,n} = 1 \) and weights \( w_n = 1 \). By doing so we have an attention mechanism similar to the one proposed in [19] but with the advantage of being able to use locations outside the window around location \( m \):

\[
\hat{f}^l(m) = \sum_{n=1}^{N} a_{m,n} f^{l-1}(m + n + \Delta m_{m,n}).
\]  (5)

This attention mechanism, unlike other attention methods, can handle fast motion without increasing the size of the window \( N \) and the computation time. Since no convolution transformation is applied, we can use it to align \( h_{t-1} \) and \( \hat{r}_{t-2:t+2} \) to \( y_t \) much faster than other techniques. Our approximation only adds two extra convolution layers, while optical-flow calculation [14] requires an entire sub-network with several convolutions.

### III. Experimental Results

The training dataset was constructed by extracting \( 10^6 \) sequences of 6 patches of size 128 x 128 pixels from the Myanmar training sequences. For each HR patch sequence we obtain its corresponding 10 LR patch sequence, so each HR patch at time \( t \) has the corresponding LR sequence of patches at time \( t-2, t-1, t, t+1, \) and \( t+2 \). To remove uninformative patches from our training dataset, patches with variance less than 0.0035 were not considered.

We train the network for 60 epochs using a batch size of 64 and sampling 10000 batches per epoch. For our recurrent networks, we first pre-train the auxiliary network \( g_\phi \) for 10 epochs. The loss we use to train our network is, instead of the Mean Squared Error (MSE), the Charbonnier loss:

\[
\gamma(\hat{x}, x) = \sum_k \sum_i \sum_j (|\hat{x}_{k,i,j} - x_{k,i,j}|^2 + \epsilon^2)^{1/2},
\]  (6)

where \( \hat{x} \) and \( x \) are the estimated and the real high-res frames respectively and \( \epsilon \) an hyper-parameter, which in our experiments is set to \( 10^{-3} \). This loss is more robust to outliers and more stable than MSE [21]. We use Adam optimizer [22] with the learning rate set to \( 10^{-3} \) for the first 20 epochs and then divided by 10 at the 20th and 40th epoch. The weight decay parameter was set to \( 10^{-3} \). We focus on upscaling factor 4 for all the experiments shown in this section.

To determine the contribution of each of the components of our proposed architecture GR-VSR, we perform an ablation study. First, we train a non-recurrent model without deformable attention but with the same depth as our model. We call this model No-R-VSR. This model is similar to VSRResNet but with half the number of residual blocks and two-step upsampling using a subpixel shuffle layer [23]. To check the contribution of the recursion, we add recursion to No-R-VSR creating a new model R-VSR. Notice that R-VSR uses neither deformable attention nor a gate. The model R-VSR-Att incorporates deformable attention to...
Fig. 2: Qualitative results of our GR-VSR model compared to current state-of-the-art methods for factor 4. In this case, GR-VSR is able to recover more details of the original high-res frame than the other non-recurrent methods. However, the results obtained by GR-VSR are less noisy and closer to the original HR image.

R-VSR. Finally, the model GR-VSR-No-Att calculates the $h_t$ using the gate network without deformable attention.

The first part of Table I shows the results of this study. All models are compared in terms of PSNR, SSIM and T-RRED[24]. The T-RRED metric is used to quantitatively measure temporal consistency (the lower the better). The most significant increase in the spatial quality metrics comes from the inclusion of the recursion, since R-VSR outperforms No-R-VSR by 0.8 dB with almost the same computational cost. As expected, the inclusion of deformable attention in R-VSR-Att further boosts the performance of the network. However, this is not the case when using only the gate network: we expected GR-VSR-No-Att to perform worse than R-VSR, since without aligning moving objects in $h_{t-1}$ and $\hat{h}_t$ have different space locations. Despite this, both GR-VSR-No-Att and R-VSR-Att have a similar performance. This indicates that the gate network in GR-VSR-No-Att has learned to only use $h_{t-1}$ information for objects that did not change location in the next time step. Finally, the gate network when used together with deformable attention can significantly increase both the quality of the frames and the temporal consistency, as shown by the difference between our full model GR-VSR and R-VSR-Att.

We now compare our GR-VSR model with the state-of-art. The second part of Table I shows this comparison. Notice that, in the case of FRVSR[14], the results are not directly comparable since the authors use a degradation different from bicubic downsampling. It can be seen that our GR-VSR significantly outperforms all the other state-of-art methods for bicubic downsampling degradation, even AVSR[9] that uses a much deeper network. This difference can be observed in the predicted frames, as shown in Fig. 2. In the case of FRVSR[14], even though our proposed network performs far less operations per time
step (8 residual blocks with 64 filters each vs 10 residual blocks with 128 filters each), we obtain comparable results. It is worth noticing that the degradation used in [14] is less aggressive than bicubic downsampling, so we can expect that our model will still slightly outperform FRVSR[14] when applied to the same degradation.

<table>
<thead>
<tr>
<th></th>
<th>PSNR</th>
<th>SSIM</th>
<th>T-RRED[24]</th>
</tr>
</thead>
<tbody>
<tr>
<td>No-R-VSR</td>
<td>25.77</td>
<td>0.7679</td>
<td>1.42</td>
</tr>
<tr>
<td>R-VSR</td>
<td>26.61</td>
<td>0.8115</td>
<td>1.37</td>
</tr>
<tr>
<td>R-VSR-Att</td>
<td>26.68</td>
<td>0.8145</td>
<td>1.34</td>
</tr>
<tr>
<td>GR-VSR-No-Att</td>
<td>26.61</td>
<td>0.8123</td>
<td>1.36</td>
</tr>
<tr>
<td>GR-VSR</td>
<td>26.76</td>
<td>0.8156</td>
<td>1.27</td>
</tr>
<tr>
<td>RCAN[20]</td>
<td>25.44</td>
<td>0.7405</td>
<td>1.71</td>
</tr>
<tr>
<td>VESPCN[4]*</td>
<td>25.35</td>
<td>0.7309</td>
<td>-</td>
</tr>
<tr>
<td>SPMC-SR[5]</td>
<td>25.63</td>
<td>0.7709</td>
<td>1.74</td>
</tr>
<tr>
<td>TAN[6]*</td>
<td>25.53</td>
<td>0.7475</td>
<td>-</td>
</tr>
<tr>
<td>VSRResNet[8]</td>
<td>25.51</td>
<td>0.7530</td>
<td>1.47</td>
</tr>
<tr>
<td>FRVSR[14]*</td>
<td>26.69</td>
<td>0.8222</td>
<td>-</td>
</tr>
<tr>
<td>AVSR[9]</td>
<td>26.17</td>
<td>0.7895</td>
<td>1.30</td>
</tr>
</tbody>
</table>

TABLE I: Results on the VidSet4 video sequences [25] for a scale factor of four in terms of PSNR, SSIM and T-RRED[24]. The first part of the table shows the ablation study for the proposed GR-VSR and the second part a comparison with current state-of-art methods. Notice that RCAN[20] is a SR method. VESPCN[4], TAN[6] and FRVSR[14] results are the ones reported in the original paper. FRVSR[14] uses a different degradation (not bicubic downsampling).

IV. CONCLUSIONS

We have introduced a new RCNN VSR model that adapts the recurrent unit GRU. Our model uses deformable attention to align the previous hidden state with the current one and a gated operation to combine them. This allows our model to better reuse features and exploit longer time relationships between the frames. The experiments show that our model, GR-VSR, outperforms current state of the art methods in terms of PSNR, SSIM and temporal consistency. Temporal consistency is naturally achieved, without the use of losses that explicitly impose temporal consistency. In the future, perceptual losses will be incorporated into the training of this model to further increase the perceptual quality of the predicted frames.

REFERENCES