Multi-Resolution Multi-Reference Talking Head Synthesis via Implicit Warping

Vibhaalakshmi Sivaraman [*⊞]   Xuan Luo [⊞]   Anne Menini [⊞]   Mohammad Alizadeh [*]   Rahul Garg [⊞]

[*]*Massachusetts Institute of Technology*, [⊞] *Google*

Abstract

Generating high-resolution frames of an individual's face from a low-resolution version is useful in a number of applications like low-bitrate video conferencing, video enhancement, and de-blurring. A common technique to improve fidelity to high-frequency content is to use one or more example or reference images at higher resolution. However, typical optical flow based models do not easily scale to multiple example images, and are constrained to generating good frames only for poses that are close to the reference pose. We propose a novel multi-resolution-attention architecture that encodes information from the reference images into a set of key-value pairs at multiple resolutions that are attended to for frame synthesis. Notably, once trained, our model can effortlessly adapt to varying numbers of reference images during inference, while during training, it efficiently requires only a single reference frame. On a highly curated dataset, we show that even with a single reference frame, our multi-resolution architecture improves the PSNR, SSIM and LPIPS of synthesized images on average by $1.71\,\mathrm{dB}$*,* $1.13\,\mathrm{dB}$ *and 0.06 respectively over a single-resolution architecture. Further, using multiple references provides an additional improvement of* $0.15\,\mathrm{dB}$ *in PSNR,* $0.07\,\mathrm{dB}$ *in SSIM, and 0.01 in LPIPS. Moreover, on a more diverse dataset, our approach exhibits a remarkable boost in reconstruction quality –* $\sim 1\,\mathrm{dB}$ *PSNR and SSIM, and* 20% *in LPIPS – when comparing the use of ten references to a single reference frame.*

1. Introduction

Generating or reconstructing faces in a novel pose using priors such as a reference image or audio information has been an area of active research the past few years [27, 22, 42, 37, 35, 19]. Most approaches use a source image, and a sparse representation of the (target) unseen pose as inputs to a neural network that has been trained to reuse relevant parts of the source image in the new pose while also synthesizing unseen portions. These sparse representations may involve keypoints [27, 42, 37], audio [50, 43, 47] or even low-resolution video input [29]. Since these sparse representations can often be compressed very efficiently, these approaches [29, 22, 37] pose new opportunities for low-bandwidth video-conferencing by trading off receiver side compute for less information transmitted per video frame.

Synthesis approaches that produce high-resolution (HR) images from low-resolution (LR) version with help from reference images have been shown to be particularly robust to extreme motion or changes in orientation [29]. Many of the existing techniques such as Gemino [29] and HIME [39], inspired by prior work on keypoint-based techniques [42, 27, 37, 29], compute optical flow in their face reconstruction pipeline. Specifically, they estimate, in two-dimensions, which source feature (or pixel) needs to be moved to a given target image location in order to produce the reconstruction. However, this estimation is challenging when there is extreme motion, and image or feature correspondence is minimal. While this can be alleviated to an extent with the use of multiple source or reference images, it often requires an additional layer [35] to compute masks that combine the different sources' optical flows. Since not all target regions can be mapped back to one or more references, these approaches rely on a partially generative network to synthesize or hallucinate such regions. Gemino [29], for instance, uses personalized generative models that capture features of a person in their weights.

A more versatile alternative to optical flow is attention [34, 5]. The basic approach within an attention layer is to compute correspondence between *query* and *key* features that identify which spatial locations in the target frame (query) should draw upon which locations in the reference (key). Each key feature maps to a *value* feature that is the actual feature used after the attention operation. Attention is similar to optical flow in that it identifies a correspondence between source and target pixels. However, it is much more general in that it allows you to produce a weighted combination of input locations for every output location, rather than rely on a single input location. This versatility comes at the cost of higher computation [5, 46] but provides much more flexibility to the model in the form of a larger corpus of features, potentially from multiple references [35], to pick from. It also opens up the ability to identify a set of person- or video-specific features that the attention mechanism can learn to draw upon appropriately in the reconstruction pipeline.

The idea of using attention in the face reconstruction pipeline has been explored recently [48, 19]. However, current approaches are either generative and may not maintain identity [48], or limit themselves to the use of fixed number of reference frames [19]. The latter approach hinders the adaptability of a single model to different numbers of reference frames without requiring costly retraining for each variation.

In this paper, motivated by the robustness of the high-frequency conditional super-resolution approach in Gemino, we discuss an alternate approach to super-resolution aided by reference frames called Gemino (Attention) that uses attention mechanisms similar to [19] instead of optical flow. Specifically, Gemino (Attention) encodes a LR target image as well as low and high-resolution versions of the reference into a set of features. It then computes attention or correspondence between the features from the LR target (query features) and the LR reference (key features), and extracts the corresponding high-resolution reference features (value features). The attended features from the high-resolution reference features are then put through a series of decoding layers to produce the final reconstruction. This attention structure naturally lends itself to multiple reference frames; we simply extract key-value pairs from each reference frame's LR and HR versions and stack them together in the attention pipeline. The model can then decide, on its own, which parts of which frame are most different and necessary to reconstruct a new target frame. Note that the model is trained once for a single reference image, but extends at inference time to multiple reference images. This is because attention is merely computing a dot product between queries and keys, and can naturally do this for more keys than it was originally trained for without retraining.Unlike [19], that only computes attention in the bottleneck layer, we compute attention at multiple resolutions and show that it improves reconstruction quality.

We evaluate Gemino (Attention) on publicly available TCDTimit [8] dataset and show that our novel multi-resolution attention design provides an improvement of $1.71\,\text{dB}$ in PSNR, $1.13\,\text{dB}$ in SSIM and 0.06 in LPIPS over a similar design that runs attention only at the coarsest level. Further, the same model when evaluated using five references provides an additional improvement of $0.15\,\text{dB}$ in PSNR, $0.07\,\text{dB}$ in SSIM, and 0.01 in LPIPS. Since TCD-Timit [8] lacks sufficient variation in pose and backgrounds, we also evalute on a more diverse dataset and observe that we get an improvement of nearly $1\,\text{dB}$ in both SSIM and PSNR, and a 20% decrease (0.01) in LPIPS when using 10 references instead of 1. As the number of reference frames grows, our model incurs higher computational costs. However, this issue can be mitigated by employing a one-time compression of the fixed key-value pairs derived from the reference frames or by utilizing faster techniques to approximate the attention mechanism [36, 26, 14].

2. Related Work

Novel-view Synthesis Approaches. Face synthesis techniques fall under the broader problem of synthesizing a novel-view of an object or a person given a certain reference view. Many proposals [49, 23] have attempted to tackle the general version of the problem including Multi-View Image Fusion [32] and NeRF [21]. These approaches typically estimate a flow from the existing views to the novel view that captures both which parts of existing reference views can be copied over to the novel view, and what parts need to be synthesized.

Approaches specific to generating *faces* in a target pose based on reference or texture information have also been widely studied [27, 42, 37, 22, 35, 19, 29, 39]. For example, the First-Order Motion Model (FOMM) [27] animates a source image of a person into the target pose by estimating the motion using a first-order approximation around a set of ten sparse keypoints that are learnt end-to-end. Maxine [37] furthers this idea by using three-dimensional keypoints. The FOMM itself has been extended to use multiple source images [35] as well as optimized to run on mobile devices [22] albeit at much lower fidelity. Since keypoints provide a limited representation of the facial movement and mouth positioning, a few approaches [50, 43, 47, 24] leverage audio data in addition to pose information. HIME [39] and Gemino [29] use a low-resolution target image as their input instead of keypoints or audio. All these approaches rely on warping the source image based on optical flow estimation between the source and target poses. A recent proposal [19] attempts to replace the flow module with an attention mechanism instead but requires a careful selection of reference images. We seek to go further by letting the model automatically use whichever and how many ever reference images amongst a pool it deems most useful.

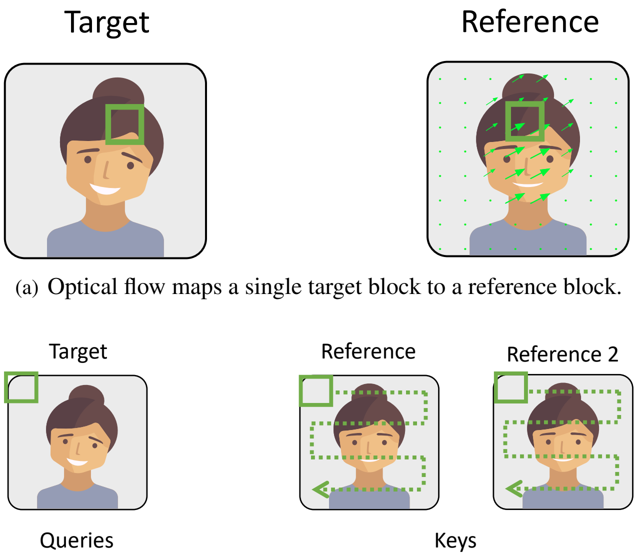
Super-resolution Techniques. Super-resolution (SR) approaches take as input a low-resolution (LR) input and reconstruct a high-resolution (HR) version. The standard form within this realm is single-image SR techniques which only use the input image. Recently, CNN-based approaches [15, 16, 18, 13] that learn this transformation have significantly outperformed classic (non-learnt) interpolation methods such as bilinear or bicubic [12]. Since a number of these models tend to be large, efforts to distill SISR models into lightweight versions have resulted in newer models such as IDN [10] and IMDN [9]. However, all of these approaches need to fundamentally hallucinate or memorize high-frequency texture information since LR inputs only capture low-frequency details.

To overcome this challenge, a handful of reference-image based SR techniques have been proposed over the years. A number of these techniques predate the neural network renaissance and use simple motion or similarity comparisons at a pixel level [3, 41, 30, 31]. More recent approaches either rely on a HR version of the LR input [45], or use learnt correspondences using patches [44, 7], features [40], or optical flow [29, 39] to extract the relevant parts of the reference image that can help reconstruct the input image. Our technique is inspired by an attention-based solution to this problem as used in [19, 40]. However, we focus exclusively on face synthesis, and provide a corpus of reference images to our model. Though our model has been trained on a single reference image like Gemino [29], at test time, it can leverage any number of reference frames and automatically combine them in the most useful way to generate the HR target image.

Attention Mechanisms. Attention mechanisms have gained popularity in vision tasks since the introduction of the self-attention layer (identical query, key, values) in [34]. A number of models [46, 17, 2] for image and video-related tasks including ViT [5] and VCT [20] have since employed either self or cross attention. Even certain super-resolution techniques rely on transformers [15, 40, 18] to help model the long-range dependencies between the low-resolution input and high-resolution target better. Recently, codebook-based approaches [33, 6] combined with attention have shown good reconstruction quality despite their use of discrete codes or quantized features that capture very specific visual parts of the frame. In particular, Codeformers [48] uses a codebook specific to faces, and uses attention to find correspondence between LR input patches and the corresponding HR patch from the codebook. However, it is a generative approach that may not synthesize images with the same identity as the HR ground-truth.

3. Motivation

As shown in Gemino [29], many of the techniques developed for novel-view synthesis using keypoints have catastrophic failures if the reference frame and the target frame differ significantly in their poses. While standard super-resolution techniques with a series of downsampling and upsampling layers has better low-frequency fidelity and places objects in the right place, it fails to capture high-frequency details such as hair strands and facial texture. Gemino [29] combines these two approaches in its high-frequency conditional super-resolution design wherein it upsamples a LR target but uses information from a high-resolution reference frame to condition the upsampling. This way, Gemino transfers high-frequency information from the reference frame to the target frame in regions that are similar, but falls back on super-resolution as a lower bound for reconstruction error for regions that are new or very different in the target frame.



(a) Optical flow maps a single target block to a reference block.

(b) Attention computes each target block (query) as a weighted average of all reference blocks (keys).

Figure 1. Comparison between optical flow and attention.

While Gemino achieves audio-like bitrates with good reconstruction fidelity, it uses only reference frame. This is because the model estimates motion between the reference and the target using optical flow methods and uses the resulting warping field to translate the reference features into the coordinate system of the target prior to decoding. The output from running optical flow maps each region or block of the target frame to its closest match in the reference frame (Fig. 1(a)). Though this is an efficient technique for motion estimation, it does not inherently let the model leverage information from multiple reference frames.

Consider the case of a target that has both eyes open and the mouth wide, but needs to choose one of two options for references: one with open eyes and another with an open mouth. An optical flow approach would only be able to leverage one of the two frames, missing out on relevant high-frequency information in either the mouth or eye region. One option would be to add a linear layer after computing individual warping fields based on each of the two references. Such a linear layer would be responsible for computing weights for which image regions need to focus on which of the two references. Such a weighing layer is outside of the default optical flow formulation. However, this is precisely what the expressiveness of an attention layer lets us do.

Attention [34] is a mechanism to capture correspondence between regions of the target frame and all regions of one or more reference frames. In contrast to optical flow which maps a region of the target to a single region in a reference frame, attention allows you to compute a particular region of the target as a weighted combination of all reference regions (Fig. 1(b)). This automatically scales to multiple references since it simply returns weights across all regions of all references. In the above example of the target with open eyes and a wide mouth, attention allows target regions around the eyes to weigh the reference with open eyes more while target regions around the mouth rely more on the other reference with the wide mouth. The precise correspondence is calculated using a region-by-region dot product between the target and the references. Since this operation does not involve any learning, an attention-based model trained on a single reference frame can be easily extended to any number of references at inference time. The model simply calculates correspondence or attention with as many references as provided before running the attended output through the rest of the decoding pipeline. Due to its versatility, we employ attention as an alternative to optical flow for motion estimation in Gemino (Attention), an alternate design to Gemino for high-frequency conditional super-resolution.

4. Method

The input to Gemino (Attention) is a set of references in both their LR and HR versions, and a LR target whose HR version we want to reconstruct. Note that the LR and HR images have the same dimensionality because the LR image is downsampled and then re-upsampled using bicubic interpolation. This results in a blurry LR image whose dimensions match that of the crisper HR image. Fig. 2 depicts the architecture of our system.

A shared *key-query encoder* encodes the LR references and the LR target into key and query features respectively. A separate *value encoder* encodes the HR references into a set of value features. An *aggregator* module (*e.g.* concatenation) aggregates key-value feature pairs from multiple references into a smaller compact set of key-value pairs. An attention layer computes attention between key and query feature pairs and maps it to the corresponding values. This process is repeated at higher resolutions (or finer levels) to provide skip connections that are leveraged during decoding to preserve high-frequency content. The attended value features at the coarsest level are then decoded, along with skip connections at subsequent levels, to produce the final reconstructed image.

We describe in detail the key-query encoder, the value encoder and decoder in §4.1. A strawman version of attention that uses a single reference image is described in §4.2. We extend this strawman to multiple references in §4.3, and finally describe how we leverage skip connections and attention at multiple resolutions to preserve high-frequency content in §4.4.

4.1. Feature Encoding and Decoding

To capture information from the LR and HR images at multiple resolutions in a richer space than their per-pixel RGB representations, we first encode them into the feature space. We then perform attention before decoding the same features.

Key-Query Encoder. The LR references and the LR version of the target frame are encoded using a shared key-query encoder. Since our attention mechanism computes correspondences between the keys and the queries, it is best that their features are semantically aligned, and so, we use a shared encoder to improve that likelihood. The key-query encoder consists of three convolutional layers each with stride 1. Each of these layers is paired with a downsampling layer that runs convolutions of stride 2. Thus, each downsampling layer reduces each spatial dimension by a factor 2, decreasing the total spatial blocks by 4x. The convolutions use a 3×3 kernel and 'same' padding to capture information across regions of the input images. Each convolutional layer also uses a ReLU activation. Fig. 3 diagrammatically describes this encoder design. Since we use 384×384 images in our evaluation, the encoder produces features with spatial dimensions of 192×192, 96×96 and 48×48.

Value Encoder. The architecture of the value encoder is the same as the key-query encoder. In other words, the value encoder also uses three convolutional layers with stride 1, each of which is paired with a downsampling layer that reduces each spatial dimensions by 2×. However, unlike the key-query encoder, the value encoder operates on high-resolution references and thus, encodes high-frequency content associated with the reference images.

Decoder. The decoder's design nearly mirrors that of the value encoder since it is designed to produce an RGB image from the attended (value) features. Specifically, the decoder consists of three convolutional layers each with stride 1. Each of these layers is paired with a two-dimensional upsampling layer that performs a bilinear interpolation to increase the spatial dimensions of the feature by 2× along each axis. The convolutions use a 3×3 kernel and 'same' padding to combine information across regions of the input images. Each convolutional layer also uses a ReLU activation. The architecture is a mirror of the encoder shown in Fig. 3 with Upsampling layers replacing the down blocks. The decoder further uses skip-connections produced by attending appropriately to the encoded key and value features at the resolutions in intermediary layers of the encoder. We motivate the use of these skip connections further in §4.4.

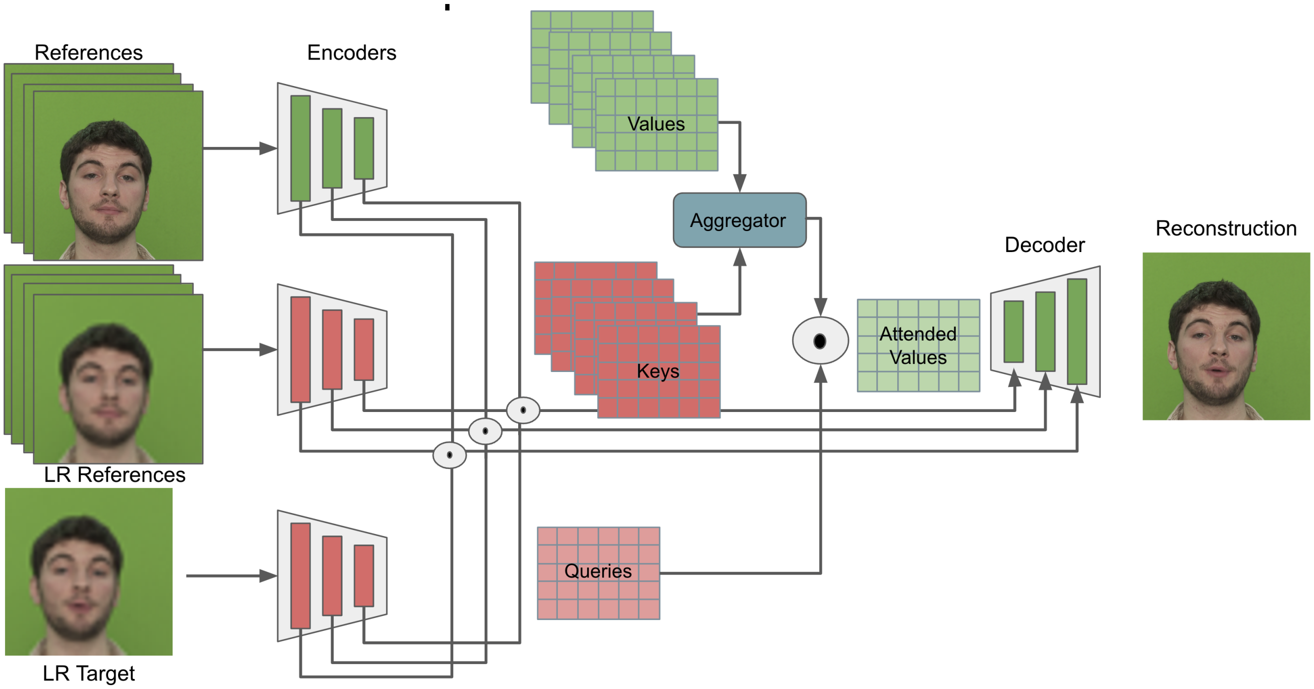


Figure 2. Gemino (Attention)'s architecture. The LR target is encoded into query features, while a set of LR and HR reference frame pairs are encoded into their respective key and value feature pairs. The key and value features can be aggregated to smaller dimensions using an aggregator (*e.g*. concatenation, clustering, PCA). Scaled dot-product attention (denoted by the '.') is computed between query features and each of the key value pairs to obtain attended HR value features that are then decoded to produce the final reconstruction.

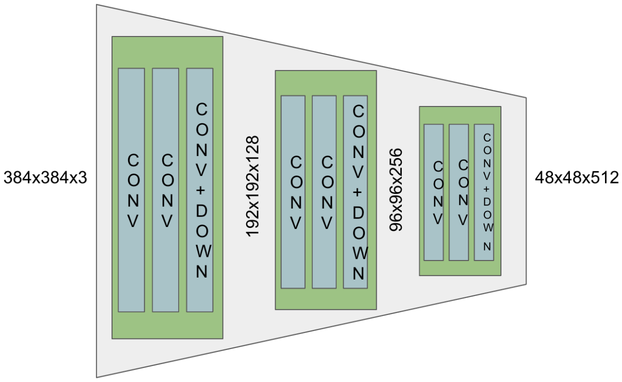


Figure 3. Gemino (Attention)'s key-query encoder architecture consisting of two convolutional blocks followed by a downsampling block (convolutions with stride 2) at each resolution. Each block uses ReLU activation. The encoding layers convert the 384×384×3 RGB image into features of size 48×48×512. The decoder architecture mirrors the encoder, but uses 2D upsampling layers instead of the downsampling layers.

4.2. Attending to a Single Reference

As described in §4.1, the key-query encoder produces a sequence of query features from the LR target frame and key features from the LR reference frames, while the value encoder encodes the HR reference frame. The next step is to compute correspondence between these query and key features. For simplicity, we assume that we have exactly one reference frame, which ensures that the query, key, and value features are all of the same dimensionality. In §4.3, we build on this design to extend it to multiple references.

To compute correspondence between the query and key features, we use the scaled dot-product attention layer [34, 1]. Specifically, we compute:

$$\text{Attended Values(Q, K, V)} = \text{softmax}\left(\frac{QK^T}{c}\right)V$$

where Q, K, and V denote query, key and value features respectively, and c is a learnt scaling parameter. This layer computes the inner-product QK^T between each query element against each key element, and obtains a softmax over it to produce a weighted average of key elements across all locations that contribute to a particular query location. Effectively, this layer captures the similarity between the query and the key or learns how to combine different regions of the reference to produce a particular region of the target. Since we ultimately want to use the HR features in the decoding procedure, this similarity, though computed on LR reference features, is mapped to HR value features from the reference via the multiplication with value features V.

If we have d dimensions in the query (assume that all spatial and channel dimensions are flattened), the query Q is $(1 \times d)$ in size. Since the key-query encoder is shared and the value encoder's architecture is identical to that of the key-query encoder, the values and the keys also have dimension d when flattened. Assuming a single reference image, this means that V and K are also $(1 \times d)$ in size. This produces a similarity value QK^T of dimensions (1×1) and attended values of dimensions $(1 \times d)$ which can then be decoded to produce the final reconstruction.

Note that we choose to use the simplest version of attention by leveraging a single dot-product attention layer with c being the only learnt parameter. Multi-headed attention [34] is more commonly used in transformer architectures, and is known to be much more powerful. However, we did not see significant gains in our evaluations, and thus chose to avoid the increased computational overheads.

4.3. Attending to Multiple References

Using attention even on a single reference image allows for improved capabilities in combining information from multiple image regions in contrast to optical flow that moves a single source location to a single target location. However, the versatility of attention is put to its true use with multiple reference images. To extend the design described in §4.2 to multiple images, we alter the attention layer to now compute correspondences across many more features from multiple references.

Specifically, we assume that we have n different reference images, and we have LR and HR versions of each reference. The key-query encoder encodes the LR references to produce key features of size $(n \times d)$ while the value encoder encodes the HR references to produce value features of size $(n \times d)$. So far, we have merely concatenated features from n different reference images. We discuss in future work possibilities to aggregate these concatenated features into a smaller representation of size $(a \times d)$ using the *aggregator* module in Fig. 2. However, we assume for the rest of this section and the evaluation, that we simply concatenate the features obtained from multiple references.

Once concatenated, the attention layer computes the same scaled-dot product attention, albeit on different dimensions. Specifically, the similarity between key and query features is now computed across all n key features to result in a value QK^T of dimensions $(1 \times n)$. However, since the attended values are $(n \times d)$ in size, the final dimensions of the attended values is still $(1 \times d)$ which can then be decoded to produce the final reconstruction.

This reveals a very powerful detail about the attention-based design in our model. The attention layer, even if trained on a single reference image to learn its scaling parameter c, can be extended at inference time to use multiple reference images *without retraining*. This is because the sizes of the key and value pairs are abstracted away in the dot-product and only the query feature dimensions and the final attended value features need to be maintained for the model pipeline from Fig. 2 to work between train and test time.

4.4. Preserving High-frequency Content

Multi-resolution Attention. The attention mechanism described above operates on query, key and value features of dimensions d. However, the actual value of d and what it means in the context of spatial and channel dimensions for the features can have an outsize impact on the granularity of these features and subsequently, the quality of the final reconstruction. A strawman solution would be to run attention once on the coarsest (last) level of features obtained from the query, key, and value features. In our evaluations with 384×384 images and encoders with three downsampling layers, this would mean spatial dimensions of 48×48. While this makes the computationally intensive attention operation tractable, such a resolution is too coarse to realistically preserve any high-frequency content associated with the images.

Instead, we leverage its skip connection design popularized by U-Nets [25] to improve the fidelity to high-frequency content in Gemino (Attention). Since we already have an encoder and decoder structure, it is relatively straightforward to extend our design to use skip connections. Specifically, we obtain encoded query, key and value features from each intermediary level of the key-query encoder and value encoder. This results in features of spatial dimensions 192×192, 96×96 and 48×48. We compute attention or correspondence between each of these levels' query and key features to produce attended value features at each intermediary level. While the coarsest level's attended features are used as the input to the decoder, the remaining levels' attended features are progressively fed in as skip connections to improve the fidelity to high-frequency content in the final reconstruction. We deem this architectural element "multi-resolution attention" and evaluate its benefits in §5.2.

Block-based Attention for Reducing Compute. Though the skip connections help with improving fidelity, running a scaled dot-product attention at resolutions as high as 192×192 is simply impractical. This is because the dot product is effectively computed between every spatial query and key location resulting in a vector of size 192×192×192×192. To overcome this issue, we break our feature space into smaller patches and compute attention only within that patch. This idea is inspired by ViT [5], but modified to account for the fact that we are only synthesizing talking heads.

In the design of Gemino (Attention), we break features that are spatially large into 12×12 blocks and only compute attention within the 144 locations inside each of these blocks and across the 144 locations in the same block of all the reference image features. In other words, query features of dimensions $192 \times 192 \times f$ where (f is the filter or channel dimension) are broken into 256 separate $12{\times}12{\times}f$ blocks with their indices identifying which region in the original $192{\times}192$ they came from. We repeat this across all n reference frames' key and value features to generate n separate key and value features for each $12{\times}12$ block index. Then, we pick a particular block index, and compute attention between the $(1 \times 144.f)$ query features and its $(n \times 144.f)$ key and value features for that block. In conceptual terms, this boils down to first narrowing down to a small set of features mapping to the eye region, and then leveraging information on the orientation of the eye across n frames to reconstruct the eye in the target. This is similarly repeated (across blocks) to attend to features in regions focused on the ear, mouth, nose, *etc*.

5. Evaluation

5.1. Setup

Dataset. We use the TCDTimit [8] dataset that consists of 1080p videos that are recordings of trained actors who read a script facing a camera against a green screen. The videos are short clips of duration less than 10 seconds and are at a framerate of $\sim$ 30 FPS. The videos are of high-quality, encoded at bitrates above $50\,\mathrm{Mbps}$. Out of the 60 actors, we choose 51 to form our training set and nine people to form our test set. Each person has 96 videos. Prior to training, we crop a square frame of size 384×384 in the center. The low-resolution (LR) input is either of size 48×48 or 96×96 depending on whether it is a 8× or 4× upsampling task. In both cases, the LR frames have been resized to 384×384 by bicubic upsampling prior to use in the training or inference pipeline. We run inference on every frame of every test video of every person, but for training, we sample 1000 pairs of reference and target frames per video. Note that the people in the train and test dataset are completely different; unlike Gemino, this approach does not leverage per-person finetuning for improved reconstruction fidelity.

Baselines. We compare the following approaches to understand the benefits that Gemino (Attention) provides compared to existing solutions.

- **Single-image Super-resolution (SISR)**: A simple encoder and decoder pair that encodes the LR target image and decodes it into the HR target image. This approach does not use any HR reference frames.
- **Gemino**: A version of the Gemino [29] model that does 8× upsampling. Gemino operates directly on the LR input (48×48) without resizing it using bicubic upsampling to 384×384. We use this only to provide a baseline for compute overheads (and not visual fidelity) since Gemino was trained on a very different dataset with personalization benefits that we do not explore in this paper.
- **Coarse Attention with Single Reference Image (Coarse Attn.)**: This approach, like Gemino, encodes the LR and HR frames into a set of encoded features. However, instead of computing optical flow to warp the HR features, it computes attention on the coarsest (or smallest spatial dimension) features of the HR frame with those of the LR frame. This attention is run over the entire spatial dimension (48×48) of these encoded features. The decoding pipeline does not use any of the skip connections described in §4.4.
- **Gemino (Attention)**: Our contribution that builds upon the previous "Coarse Attention" model by running attention at multiple resolutions using the intermediary outputs of the encoder's layers when applied to the HR and LR reference and target frames. Each of these intermediary attended outputs is provided to the decoder as skip connections to improve its high-frequency fidelity. To ensure that this approach scales as the number of reference frames is increased, attention is computed over smaller blocks to focus the attention mechanism over small regions of all the references. This model is trained using a single reference image randomly sampled from the same video as the target image, but is used at inference time with upto 10 reference images in our evaluations. In such cases, the encoded features from all the references are provided as key-value pairs to the attention module that decided how to appropriately weigh them when computing features to decode into the target image.

Model and Training Procedure. We train all models on the entire corpus of train videos for 1M steps. Gemino (Attention) is trained with a batch size of 8 with random rotational augmentations of ninety degrees to improve its robustness. The learning rate starts at 0.0001 and decays by a factor of 0.46 halfway. Our model is trained with both an L1-loss on the pixel space between the ground truth and the reconstruction, as well as a feature matching loss from VGG [11] that improves the sharpness of the output produced. Both these losses are weighed equally.

Visual Quality Metrics. We compute the average Peak Signal-to-Noise Ratio (PSNR), the Structural Similarity Index (SSIM) [38], and the Learned Perceptual Image Patch Quality Metrics (LPIPS) on the reconstructed frames from each of the above baselines and the ground-truth. PSNR is reported as the mean-square error of pixels in each frame averaged across all test frames of a given video, measured on the RGB scale. SSIM is reported as the average (across frames) of the structural similarity between the reconstruction and its high-resolution counter-part. For SSIM and PSNR, a higher value denotes better reconstruction quality. LPIPS is reported as a distance metric in the VGG [28] feature spaces of the reconstructed image and its high-resolution ground-truth. Lower LPIPS scores correspond to higher fidelity. We also show visuals to provide qualitative

	PSNR (dB) ↑			SSIM (dB) ↑			LPIPS ↓		
Scheme	Avg.	P5	P1	Avg.	P5	P1	Avg.	P95	P99
SISR	30.98	28.72	28.56	8.14	6.77	6.69	0.17	0.21	0.22
Coarse Attention	30.98	28.73	28.55	8.07	6.70	6.62	0.17	0.20	0.21
Gemino (Attention): 1 ref	32.69	29.86	29.39	9.55	7.19	7.04	0.11	0.15	0.16
Gemino (Attention): 2 refs	32.67	29.83	29.42	9.50	7.25	7.08	0.11	0.15	0.16
Gemino (Attention): 5 refs	**32.84**	**29.86**	**29.46**	**9.62**	**7.33**	**7.13**	**0.10**	**0.14**	**0.15**

Table 1. Performance improvements from using multiple references with Gemino (Attention) running over 24×24 blocks over Single-Image SR (SISR) and attention that runs at the coarsest level alone (Coarse Attention) at an 8× upsampling task. SISR and Coarse Attention miss high-frequency information while Gemino (Attention) retains it.

comparisons across approaches.

Computational Overhead Metrics. In addition to assessing the fidelity of the reconstructions to their ground truth, we also measure the computational overhead of each of the models. This is important because attention is an expensive operation since it computes large matrix dot products. We measure the the size of the model in terms of its number of parameters, its computational complexity as reflected by the number of floating point operations (FLOPS) it needs, as well as the time it takes to run inference on a single frame on a V100 GPU. We ignore any one-time operations such as encoding time of the references for Gemino (Attention) since this will likely be done once, and then reused for subsequent attention computations.

5.2. Results

Main Takeaway. We compare all of the baselines against Gemino (Attention) when upsampling 8× from a 48×48 frame to a 384×384 frame and present a quantitative summary in Tab. 1. We show visual samples in Figures 4 and 5 to provide qualitative examples. Overall, Gemino (Attention) outperforms other baselines, and also provides benefits in reconstruction quality as the number of references increases. Specifically, it maintains high-frequency fidelity when compared to SISR and Coarse Attention, but also performs better in particular regions of the face and torso with improved information from multiple references. We break these results down in subsequent paragraphs.

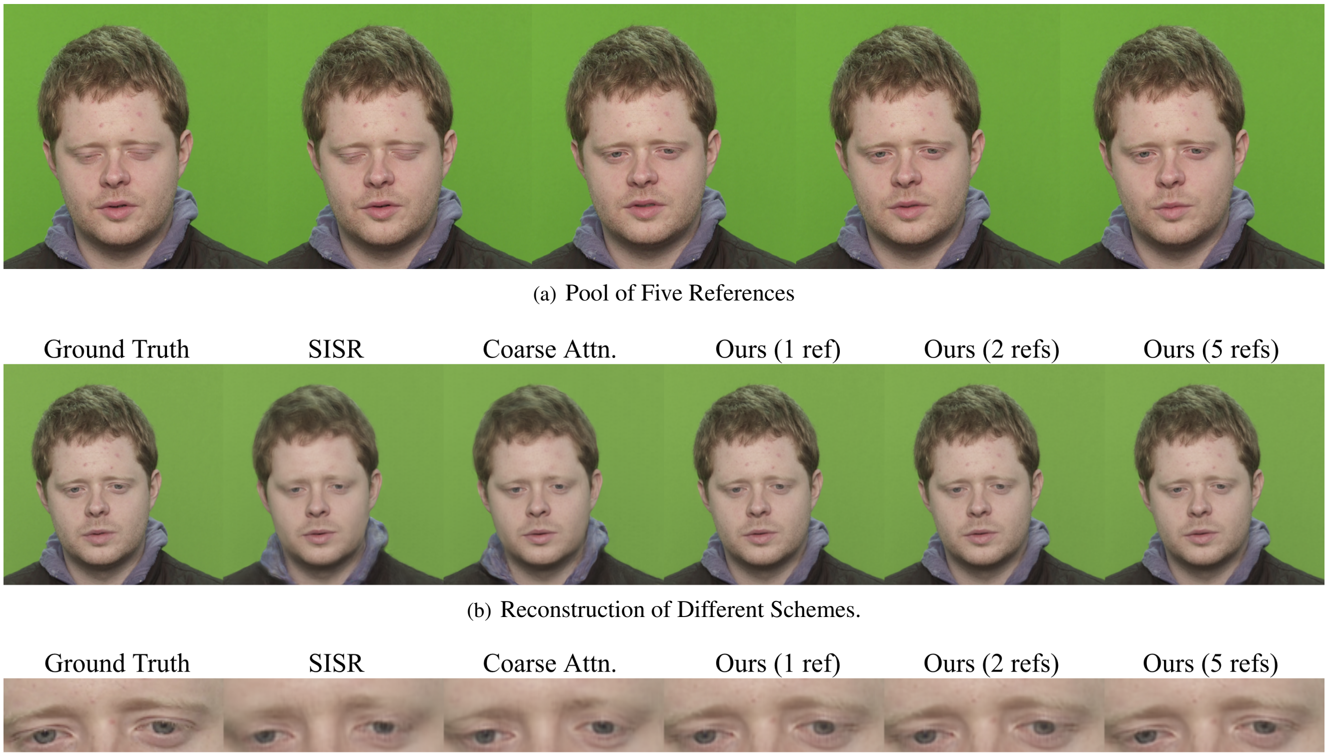
Comparison with Single-Image SR. We observe in Tab. 1 that single-image SR (SISR) achieves on average 1.7 dB less PSNR, 1.41 dB less SSIM, and 0.06 more LPIPS than Gemino (Attention) with one reference frame. A similar difference of 0.06 manifests with LPIPS on the worst 5% and 1% of test frames. Without a reference frame to provide high frequency information, SISR's upsampling layers are unable to recover the details associated with individuals' facial features. As seen in Figures 4 and 5, the SISR output misses the freckles and acne associated with both individuals' foreheads. Its reconstruction of both individuals' hair also lacks detail. This is unsurprising since the low-resolution input frame lacks high-frequency detail. Further, these models have not been trained with frames of the test individuals and thus, cannot encode such information into their weights.

Benefits of Multi-resolution Attention. Tab. 1 shows that "Coarse Attention" does not perform better than SISR. This suggests that running attention at the coarsest level with encoded reference features fails to capture the same high-frequency details that SISR misses. This is reflected in the visual samples in Figures 4 and 5 too; Coarse Attention misses the same freckles and acne that SISR misses on the foreheads. This is because the encoded features at the coarsest level of the encoder are too low-resolution (48×48) to retain enough high-frequency information from the reference frame that can later be leveraged through attention before the decoding pipeline.

However, when that same information is passed through a series of skip connections in Gemino (Attention), the reconstruction quality improves significantly even with one reference frame. Specifically, the skip connections provide reference features at multiple resolutions via attention computed on each intermediary encoder layer's outputs. The decoder then uses these reference features at the appropriate resolution to recover the high-frequency details associated with the individual. This improved architecture preserves the acne and freckles in the visual samples produced by Gemino (Attention) in Figures 4 and 5.

SR Task Difficulty. In Tab. 2, we compare existing baselines to Gemino (Attention) at a 4× upsampling task from 96×96 to 384×384. Like the 8× upsampling case, we observe that Gemino (Attention) outperforms SISR and attention at the coarsest level alone due to the lack of sufficient high-frequency information in both approaches. As the number of references is increased to ten, we observe a marginal improvement in visual quality as reported by the PSNR, SSIM and LPIPS values for the average and worst 5% and 1% of frames. Since 4× upsampling is generally easier than 8×, all of the reported numbers are much better than those reported in Tab. 1.

Computational Overheads. Though using attention with



(a) Pool of Five References

(b) Reconstruction of Different Schemes.

(c) Zoomed-In View of Reconstruction of Different Schemes.

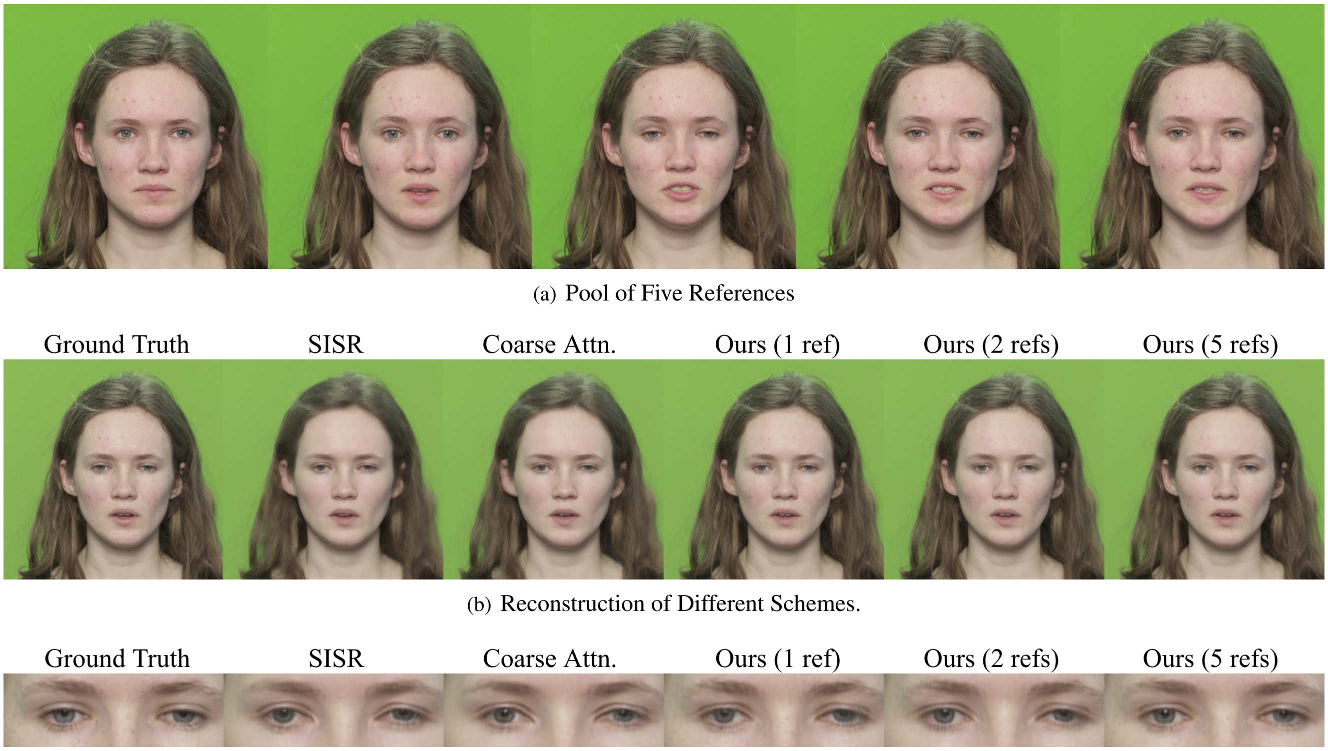
Figure 4. Reconstruction quality of different approaches on a specific frame using upto five references. Unlike Gemino (Attention), SISR and Coarse Attention miss high-frequency information associated with freckles and acne on the face. As the number of references for Gemino (Attention) is increased, the reconstruction of the eyelashes and the folds of the eyes improves as seen in the zoomed-in version.

	PSNR (dB) ↑			**SSIM (dB) ↑**			**LPIPS ↓**		
Scheme	Avg.	P5	P1	Avg.	P5	P1	Avg.	P95	P99
SISR	33.61	31.23	31.02	9.76	8.10	8.02	0.12	0.14	0.15
Coarse Attention	33.61	31.18	30.97	9.78	8.16	8.05	0.12	0.14	0.15
Gemino (Attention): 1 ref	34.57	32.01	31.31	10.77	8.54	8.40	0.09	0.11	0.13
Gemino (Attention): 2 refs	34.55	32.04	31.31	10.73	8.60	8.44	0.09	0.11	0.13
Gemino (Attention): 5 refs	34.63	**32.05**	31.32	10.78	8.64	8.44	0.09	0.11	0.13
Gemino (Attention): 10 refs	**34.71**	31.98	**31.32**	**10.86**	**8.65**	**8.46**	**0.09**	**0.11**	**0.13**

Table 2. Performance improvements from using multiple references (15 frames apart) with Gemino (Attention) running over 12×12 blocks over Single-Image SR (SISR) and attention that runs at the coarsest level alone (Coarse Attention) at an 4× upsampling task. The trends across approaches are similar to 8× upsampling but the reconstruction quality is much better since the starting resolution is higher.

multiple references instead of optical flow improves the visual quality of generated frames, computing attention involves multiplication of large matrices that imposes severe overheads. As seen in Tab. 3, single-image SR takes only $11.65\,\mathrm{ms}$ to run inference on a 384×384 frame, and has far fewer parameters and FLOPs in comparison to its attention counterparts. Gemino [29] has a lot more paramters ($\sim$ 82M) because it has a separate keypoint extraction and motion estimation pipeline. However, because many of these operations run at lower resolutions in its multi-scale architecture, this does not cause an explosive increase in FLOPs or inference time.

All of the attention versions of Gemino and coarse attention have fewer parameters than Gemino's optical flow version because we use dot-product scaled attention for motion estimation which only has one parameter regardless of the attention block size and the input dimensions. However, as the block size increases for a fixed number of references, the FLOPs for Gemino (Attention) increases because each attention operation computes matrix multiplication over larger spatial regions . This consequently manifests as an increase in V100 inference time. As the number



(a) Pool of Five References

(b) Reconstruction of Different Schemes.

(c) Zoomed-In View of Reconstruction of Different Schemes.

Figure 5. Reconstruction quality of different approaches on a specific frame using upto five references. Unlike Gemino (Attention), SISR and Coarse Attention miss high-frequency information associated with freckles and acne on the face. As the number of references for Gemino (Attention) is increased, the reconstruction of the eyelashes and the eye gaze improves as seen in the zoomed-in version.

Scheme	**Params**	**FLOPs**	**V100 Inference**
SISR	11.66M	191B	11.65ms
Coarse Attn.	18.60M	385B	285.81ms
Gemino	82.41M	82B	14.87 ms
Gemino (Attention) @ 12×12			
1 ref	20.22M	481B	29.72ms
2 refs	20.22M	655B	30.15ms
5 refs	20.22M	1178B	47.38ms
10 refs	20.22M	2049B	66.14ms
Gemino (Attention) @ 24×24			
1 ref	20.22M	540B	31.67ms
2 refs	20.22M	774B	47.23ms
5 refs	20.22M	1476B	98.70ms

Table 3. Computational overheads of Gemino (Attention) at block sizes of 12×12 and 24×24 with different number of reference frames in comparison to single-image SR, attention at the coarsest level and Gemino's optical flow version. Parameters, floating point operations (FLOPS), and inference time on a V100 GPU are measured in millions, billions, and milliseconds respectively.

of references increases, attention becomes more expensive (increased FLOPs and inference time) once again because of more matrix multiplications. Note that this increase is not quite linear because we only measure the inference time for the attention and decoding pipeline, and assume that the encoding process from multiple references into features is only done once at the beginning of the video call.

Since attention at the coarsest level computes attention over 48× 48 blocks, its inference time is quite high due to prohibitively large matrix multiplication operations despite the fact that the encoded features themselves are only 48×48 (only one attention block as a result). Note that the FLOPs for the coarse attention approach is lower than Gemino (Attention) running attention over multiple separate 12×12 or 24×24 blocks. Yet, the size of the attention matrix with coarse attention at 48×48 itself is 16 times larger than Gemino (Attention) with 24×24 blocks. This could result in more memory bottlenecks [4] when creating large matrices causing the overall inference time with coarse attention to be nearly 3× larger than Gemino (Attention).

5.3. Gemino (Attention) Ablation

Number of Reference Frames in Gemino (Attention). Tab. 1 shows small improvements in reconstruction quality for the average and worst 5% and 1% of frames as the number of references in Gemino (Attention) is in-

Scheme	PSNR (dB) ↑ Avg.	PSNR (dB) ↑ P5	PSNR (dB) ↑ P1	SSIM (dB) ↑ Avg.	SSIM (dB) ↑ P5	SSIM (dB) ↑ P1	LPIPS ↓ Avg.	LPIPS ↓ P95	LPIPS ↓ P99
Gemino (Attention) w/ consecutive references	32.74	**29.87**	29.40	9.62	7.23	7.06	0.11	0.15	0.16
Gemino (Attention) w/ references 15 frames apart	**32.84**	29.86	**29.46**	**9.62**	**7.33**	**7.13**	**0.10**	**0.14**	**0.15**

Table 4. Impact of space between five reference frames in Gemino (Attention). Larger inter-frame gaps leads to more diverse reference frames in comparison to using consecutive frames that are likely to be very similar and consequently leads to better performance.

Scheme	PSNR (dB) ↑ Avg.	PSNR (dB) ↑ P5	PSNR (dB) ↑ P1	SSIM (dB) ↑ Avg.	SSIM (dB) ↑ P5	SSIM (dB) ↑ P1	LPIPS ↓ Avg.	LPIPS ↓ P95	LPIPS ↓ P99
Gemino (Attention) @ 6×6	32.30	29.33	28.97	9.13	7.37	6.99	0.12	0.16	0.18
Gemino (Attention) @ 12×12	32.21	29.33	28.90	9.04	7.16	6.89	0.12	0.15	0.18
Gemino (Attention) @ 24×24	**32.84**	**29.86**	**29.46**	**9.62**	**7.33**	**7.13**	**0.10**	**0.14**	**0.15**

Table 5. Impact of attention block size in Gemino (Attention) when using five references. A larger block size allows the attention module to leverage information across a wider region when computing correspondence, leading to better performance.

creased. Specifically, we observe 0.15 dB increase in PSNR, 0.07 dB increase in SSIM and 0.01 decrease in LPIPS when using five references. These quantitative differences are small because the TCD-Timit dataset [8] is highly curated to involve actors speaking a script in solely front-facing poses with minimal movement. We anticipate the differences being more substantial in datasets that involve more movement and a range of poses.

Despite these small quantitative differences, we observe visual differences with five references in Figures 4(c) and 5(c). Specifically, Fig. 4(c) shows that the reconstruction of the eyelashes and the folds of the eyes is a lot more accurate with five references where the latter three have open eyes when compared to using one or two references with closed eyes. Similarly, Fig. 5(c) shows a more accurate reconstruction of the eye gaze and lashes with five reference frames that have different gazes than with two references with very similar gazes.

Type of Reference Frames in Gemino (Attention). To understand whether the underlying diversity within the reference frames also impacts the reconstruction quality in addition to the number of reference frames, we choose five reference frames with two different strategies. Specifically, we choose five consecutive reference frames that are likely to be very similar and compare that against using five reference frames at intervals of fifteen frames from the origin test video, and compare the resulting output. Tab. 4 compares the two approaches and suggests that picking reference frames that are fifteen frames apart leads to better visual quality for the average and worst frames. We suspect that this is because frames that are further apart in the underlying video are likely to be more different and lend high-frequency information from different parts of the face that can be leveraged for better fidelity in the final output.

Impact of Attention Block Size. We show the effect that the attention block size has on reconstruction quality with five references when upsampling by 8× in Tab. 5. The attention block size determines the size of the region over which attention is computed or how large of a reference frame area is used to compute weights for a particular target location. Tab. 5 suggests that as the attention block size increases, the reconstruction quality (LPIPS) improves for the average and worst frames in the corpus. As we move from 6×6 blocks to 24×24 blocks, we observe an improvement of 0.54 dB in PSNR, 0.49 dB in SSIM and 0.02 in LPIPS for the average frame. Though we anticipate some improvements from increased reference information at higher block sizes, attention with 48×48 blocks becomes prohibitively expensive to the point of not being able to incorporate multiple reference frames. At such large blocks, the attention weights also tend to average out more causing blurring effects similar to “Coarse Attention”. As a result, for 8× upsampling, we use 24×24 blocks.

5.4. Results on a More Diverse Dataset

To show the benefits from multiple references on a more diverse dataset, we evaluate our ideas on a different dataset consisting of individuals in front-facing positions. Unlike TCDTimit [8], this dataset does not consist of actors with scripts with minimal head movement. In contrast, the individuals in this harder dataset tend to move around more and exhibit variety in their poses. Their backgrounds also differ across videos and do not consist of a single green screen. However, the videos are not of such high quality

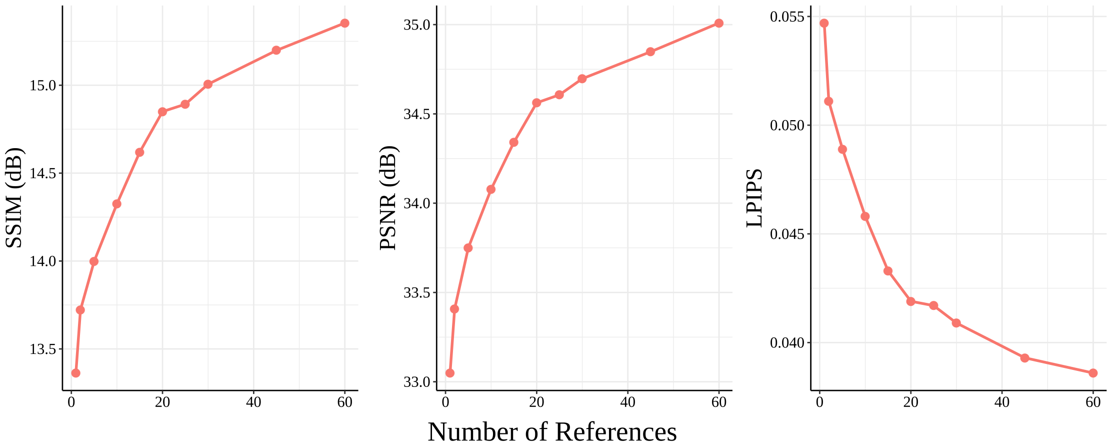


Figure 6. Visual quality on a more diverse dataset with increasing number of references. We see diminishing returns as the number of references is increased, but over 1 dB improvement in PSNR and 20% reduction in LPIPS in going from 1 to even 10 references.

and tend to be close to a few Mbps in their bitrates. In an effort to show results at higher numbers of references ($\sim$ 60), we only use the "Coarse Attention" mechanism without multi-resolution attention or the skip connections described in §4.4. This is because running attention at multiple resolutions incurs prohibitively high memory overheads as the number of references is increased.

Fig. 6 shows the improvement in visual quality on the above described dataset as the number of references is increased when upsampling 8× to a final resolution of 384×384. Specifically, on this dataset, we observe a nearly 1 dB improvement in both SSIM and PSNR and a 20% decrease (0.01) in LPIPS in going from 1 to even 10 references. This is in contrast to the 4× upsampling case with TCDTimit shown in Tab. 2 that sees very slight improvements in going from 1 to 10 references. Further, we see diminishing returns from increasing the references all the way to 60 with this dataset. While there is a big improvement in going from 1 to about 20 references, subsequent increases from 20 to 30 or 45 references show far less improvement in video quality. Note that these reference frames are consecutive and we run attention only at one-resolution, yet there are significant improvements from using more references when compared to TCDTimit [8]. We anticipate more gains with attention at multiple resolutions and reference frames that are further apart.

6. Limitations and Future Work

Though Gemino (Attention) addresses the issue of being constrained to a single reference frame in Gemino, the solution comes at significant compute costs. Specifically, the attention-based architecture involves many more matrix multiplication operations on large multi-dimensional matrices. This results in an increase in FLOPs and consequently, inference time as the number of references is increased. However, unlike Gemino, its attention counterpart has not been optimized to use depthwise-convolutions, knowledge distillation or minimal number of kernels and filters for optimal reconstruction. These standard model compression techniques can be applied to Gemino (Attention) as well to improve its reconstruction time across the hardware spectrum. Recent developments such as FlashAttention [4] can further be employed to align the attention mechanism with IO bottlenecks in a way that reduces memory overheads. Additionally, methods such as those discussed in [36, 14, 26] can be leveraged to compress the key-value pairs calculated from reference frames or to create approximations of the attention mechanism. Given that reference frames tend to remain fixed throughout the entire inference process and often contain substantial redundancy, both compression and approximation techniques are highly relevant in this context.

7. Conclusion

In this paper, we develop an alternate design for the high-frequency conditional super-resolution technique proposed by Gemino [29] that leverages attention for motion estimation. Our approach is able to utilize high-frequency information from a diverse set of references to better predict the target frame than with one single reference. The scaled-dot product attention layer also allows flexibility in picking the exact number of references at inference time even though the model is trained for only one reference frame. This flexibility allows for the design of new adaptation algorithms that dynamically choose the number and specific reference frames that trade off accuracy for compute when deployed on the receiver side in settings such as video conferencing. We leave the design of such algorithms as well as optimizations to speed up our attention-based model architecture to future work.

References

[1] Keras Attention Layer. https://keras.io/api/layers/attention_layers/attention/. 5

[2] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, volume 2, page 4, 2021. 3

[3] Vivek Boominathan, Kaushik Mitra, and Ashok Veeraraghavan. Improving resolution and depth-of-field of light field cameras using a hybrid imaging system. In *2014 IEEE International Conference on Computational Photography (ICCP)*, pages 1–10, 2014. 3

[4] Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness, 2022. 10, 12

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1, 2, 3, 6

[6] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 3

[7] Lei Han Ziwei Xu Haoqian Wang Yebin Liu Haitian Zheng, Mengqi Ji and Lu Fang. Learning cross-scale correspondence and patch-based synthesis for reference-based. super-resolution. In Gabriel Brostow Tae-Kyun Kim, Stefanos Zafeiriou and Krystian Mikolajczyk, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 138.1–138.13. BMVA Press, September 2017. 3

[8] Naomi Harte and Eoin Gillen. Tcd-timit: An audio-visual corpus of continuous speech. *IEEE Transactions on Multimedia*, 17(5):603–615, 2015. 2, 7, 11, 12

[9] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *Proceedings of the 27th acm international conference on multimedia*, pages 2024–2032, 2019. 3

[10] Zheng Hui, Xiumei Wang, and Xinbo Gao. Fast and accurate single image super-resolution via information distillation network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 723–731, 2018. 3

[11] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 7

[12] Robert Keys. Cubic convolution interpolation for digital image processing. *IEEE transactions on acoustics, speech, and signal processing*, 29(6):1153–1160, 1981. 3

[13] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654, 2016. 2

[14] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In *ICLR*, 2020. 2, 12

[15] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1833–1844, 2021. 2, 3

[16] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017. 2

[17] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 3

[18] Zhisheng Lu, Juncheng Li, Hong Liu, Chaoyan Huang, Linlin Zhang, and Tieyong Zeng. Transformer for single image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 457–466, 2022. 2, 3

[19] Arun Mallya, Ting-Chun Wang, and Ming-Yu Liu. Implicit Warping for Animation with Image Sets. In *NeurIPS*, 2022. 1, 2, 3

[20] Fabian Mentzer, George Toderici, David Minnen, Sung-Jin Hwang, Sergi Caelles, Mario Lucic, and Eirikur Agustsson. Vct: A video compression transformer. *arXiv preprint arXiv:2206.07307*, 2022. 3

[21] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM*, 65(1):99–106, dec 2021. 2

[22] Maxime Oquab, Pierre Stock, Daniel Haziza, Tao Xu, Peizhao Zhang, Onur Celebi, Yana Hasson, Patrick Labatut, Bobo Bose-Kolanu, Thibault Peyronel, et al. Low bandwidth video-chat compression using deep generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2388–2397, 2021. 1, 2

[23] Eunbyung Park, Jimei Yang, Ersin Yumer, Duygu Ceylan, and Alexander C. Berg. Transformation-grounded image generation network for novel 3d view synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2

[24] Yurui Ren, Ge Li, Yuanqi Chen, Thomas H. Li, and Shan Liu. Pirenderer: Controllable portrait image generation via semantic neural rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13759–13768, October 2021. 2

[25] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. 6

[26] Aurko Roy, Mohammad Saffar, Ashish Vaswani, and David Grangier. Efficient content-based sparse attention with routing transformers. In *TACL*, 2020. 2, 12

[27] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. *Advances in Neural Information Processing Systems*, 32, 2019. 1, 2

[28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 7

[29] Vibhaalakshmi Sivaraman, Pantea Karimi, Vedantha Venkatapathy, Mehrdad Khani, Sadjad Fouladi, Mohammad Alizadeh, Frédo Durand, and Vivienne Sze. Gemino: Practical and robust neural compression for video conferencing, 2022. 1, 2, 3, 7, 9, 12

[30] Libin Sun and James Hays. Super-resolution from internet-scale scene matching. In *2012 IEEE International conference on computational photography (ICCP)*, pages 1–12. IEEE, 2012. 3

[31] Radu Timofte, Vincent De Smet, and Luc Van Gool. Anchored neighborhood regression for fast example-based super-resolution. In *Proceedings of the IEEE international conference on computer vision*, pages 1920–1927, 2013. 3

[32] Marc Comino Trinidad, Ricardo Martin Brualla, Florian Kainz, and Janne Kontkanen. Multi-view image fusion. 2019. 2

[33] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 3

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 1, 3, 4, 5, 6

[35] Anna Volokitin, Stefan Brugger, Ali Benlalah, Sebastian Martin, Brian Amberg, and Michael Tschannen. Neural face video compression using multiple views, 2022. 1, 2

[36] Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity, 2020. 2, 12

[37] Ting-Chun Wang, Arun Mallya, and Ming-Yu Liu. One-shot free-view neural talking-head synthesis for video conferencing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10039–10049, June 2021. 1, 2

[38] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 7

[39] Xiaoyu Xiang, Jon Morton, Fitsum A Reda, Lucas D Young, Federico Perazzi, Rakesh Ranjan, Amit Kumar, Andrea Colaco, and Jan P Allebach. Hime: Efficient headshot image super-resolution with multiple exemplars. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1694–1704, 2023. 1, 2, 3

[40] Fuzhi Yang, Huan Yang, Jianlong Fu, Hongtao Lu, and Baining Guo. Learning texture transformer network for image super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5791–5800, 2020. 3

[41] Huanjing Yue, Xiaoyan Sun, Jingyu Yang, and Feng Wu. Landmark image super-resolution by retrieving web images. *IEEE Transactions on Image Processing*, 22(12):4865–4878, 2013. 3

[42] Egor Zakharov, Aleksei Ivakhnenko, Aliaksandra Shysheya, and Victor S. Lempitsky. Fast bi-layer neural synthesis of one-shot realistic head avatars. *CoRR*, abs/2008.10174, 2020. 1, 2

[43] Zhimeng Zhang, Lincheng Li, Yu Ding, and Changjie Fan. Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3661–3670, June 2021. 1, 2

[44] Zhifei Zhang, Zhaowen Wang, Zhe Lin, and Hairong Qi. Image super-resolution by neural texture transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7982–7991, 2019. 3

[45] Haitian Zheng, Mengqi Ji, Haoqian Wang, Yebin Liu, and Lu Fang. Crossnet: An end-to-end reference-based super resolution network using cross-scale warping. In *Proceedings of the European conference on computer vision (ECCV)*, pages 88–104, 2018. 3

[46] Daquan Zhou, Bingyi Kang, Xiaojie Jin, Linjie Yang, Xiaochen Lian, Zihang Jiang, Qibin Hou, and Jiashi Feng. Deepvit: Towards deeper vision transformer. *arXiv preprint arXiv:2103.11886*, 2021. 2, 3

[47] Hang Zhou, Yasheng Sun, Wayne Wu, Chen Change Loy, Xiaogang Wang, and Ziwei Liu. Pose-controllable talking face generation by implicitly modularized audio-visual representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4176–4186, June 2021. 1, 2

[48] Shangchen Zhou, Kelvin CK Chan, Chongyi Li, and Chen Change Loy. Towards robust blind face restoration with codebook lookup transformer. *arXiv preprint arXiv:2206.11253*, 2022. 2, 3

[49] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A Efros. View synthesis by appearance flow. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 286–301. Springer, 2016. 2

[50] Yang Zhou, Xintong Han, Eli Shechtman, Jose Echevarria, Evangelos Kalogerakis, and Dingzeyu Li. Makelttalk: Speaker-aware talking-head animation. 39(6), nov 2020. 1, 2