Supplementary Material for
Unpaired Image-to-Speech Synthesis with Multimodal Information Bottleneck

Shuang Ma
SUNY Buffalo
Buffalo, NY
shuangma@buffalo.edu

Daniel McDuff
Microsoft
Redmond, WA
damcduff@microsoft.com

Yale Song
Microsoft
Redmond, WA
yalesong@microsoft.com

1. Network Architectures and Parameter Settings

We provide implementation details of our model with the parameter settings used in our experiments. We encourage the readers to refer to Figure 3 and Figure 4 of our main paper when reading this section. We use the following notations to refer to commonly used computation blocks in the neural networks: Conv1D(#channels, kernel_size, stride_size), Conv2D(#channels, kernel_size, stride_size), FC(#units), GRU(#units). $\oplus(\cdot)_{res}$ refers to the residual connection. We use the superscript j to refer to the modalities $j \in \{img, txt, spch\}$.

1.1. Encoders (Figure 3 (left) in the main paper)

- **Image encoder:** $\mathbf{x}^{img} \rightarrow$ Conv2D(64, 4, 2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv2D(128, 4, 2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv2D(256, 4, 2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv2D(512, 4, 2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ MaxPool $\rightarrow \mathbf{e}^{img}$
- **Text encoder:** $\mathbf{x}^{txt} \rightarrow$ LookupTable(66, 128) $\rightarrow$ FC(256) $\rightarrow$ ReLU $\rightarrow$ Dropout(0.5) $\rightarrow$ FC(128) $\rightarrow$ ReLU $\rightarrow$ Dropout(0.5) $\rightarrow$ CBHG [2] $\rightarrow$ AvgPool $\rightarrow$ FC(512) $\rightarrow$ tanh $\rightarrow \mathbf{e}^{txt} \in \mathbb{R}^{512}$
- **Speech encoder:** $\mathbf{x}^{spch} \rightarrow$ Conv2D(32, 3, 2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv2D(64, 3, 2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ GRU(256) $\rightarrow$ FC(512) $\rightarrow$ tanh $\rightarrow \mathbf{e}^{spch}$

1.2. Multimodal Information Bottleneck (Figure 4 in the main paper)

- **Modality transformer:** $\mathbf{e}^j \rightarrow$ FC(256) $\rightarrow$ ReLU $\rightarrow \oplus\Big($ Conv1D(256, 1, 1) $\rightarrow$ ReLU $\rightarrow$ BN $\rightarrow$ Conv1D(256, 1, 1) $\rightarrow$ ReLU $\rightarrow$ BN$\Big)_{res} \rightarrow$ tanh $\rightarrow \mathbf{z}^j$
- **Memory fusion module:**
  1. Define: Memory $M \in \mathbb{R}^{n_k \times d_k / n_{heads}}$, where $n_k = 128, d_k = 256, n_{heads} = 4$
  2. Query $\mathbf{q}^j\mathbf{z}^j$, Key $\mathbf{k}$ Conv1D(256, 1, 1)(tanh(M)), Value $\mathbf{v}$ tanh(M)
  3. $(\mathbf{q}_h^j, \mathbf{k}_h, \mathbf{v}_h)$SplitHeads$(\mathbf{q}^j, \mathbf{k}, \mathbf{v}), h = 1, \cdots, n_{heads}$
  4. α_h^jSoftMax $\left(\mathbf{q}_h^j\mathbf{k}_h/\sqrt{d_k}\right), h = 1, \cdots, n_{heads}$
  5. $\mathbf{u}_h^j\alpha_h^j \times \mathbf{v}_h, h = 1, \cdots, n_{heads}$
  6. $\mathbf{u}^j$ ConcatHeads$(\mathbf{u}_h^j)$

1.3. Decoders (Figure 3 (right) in the main paper)

- **Image decoder:** $\mathbf{u}^{txt} \rightarrow$ Conv2D$^\intercal$(32, 4, 2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv2D$^\intercal$(16, 4, 2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv2D$^\intercal$(8, 4, 2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv2D$^\intercal$(8, 4, 2) $\rightarrow$ tanh $\rightarrow \mathbf{y}^{img}$
- **Text decoder:** $\mathbf{u}^{img} \rightarrow$ Dropout(LSTM(128), 0.3) $\rightarrow$ Dropout(LSTM(128), 0.3) $\rightarrow$ FC($n_{symbols}$) $\rightarrow$ SoftMax $\rightarrow \mathbf{y}^{txt}$
- **Speech decoder:** $\mathbf{u}^{txt} \rightarrow$ AttentionRNN(GRU 256) $\rightarrow$ DecoderRNN6(Dropout(LSTM(256), 0.3) $\rightarrow$ Dropout(LSTM(256), 0.3)) $\rightarrow$ reshape $\rightarrow \mathbf{y}^{mel-spectrogram} \rightarrow$ CBHG [2] (80 mels) $\rightarrow$ FC(1025) $\rightarrow$ Griffin-Lim ($\mathbf{y}^{linear}$) $\rightarrow \mathbf{y}^{speech}$

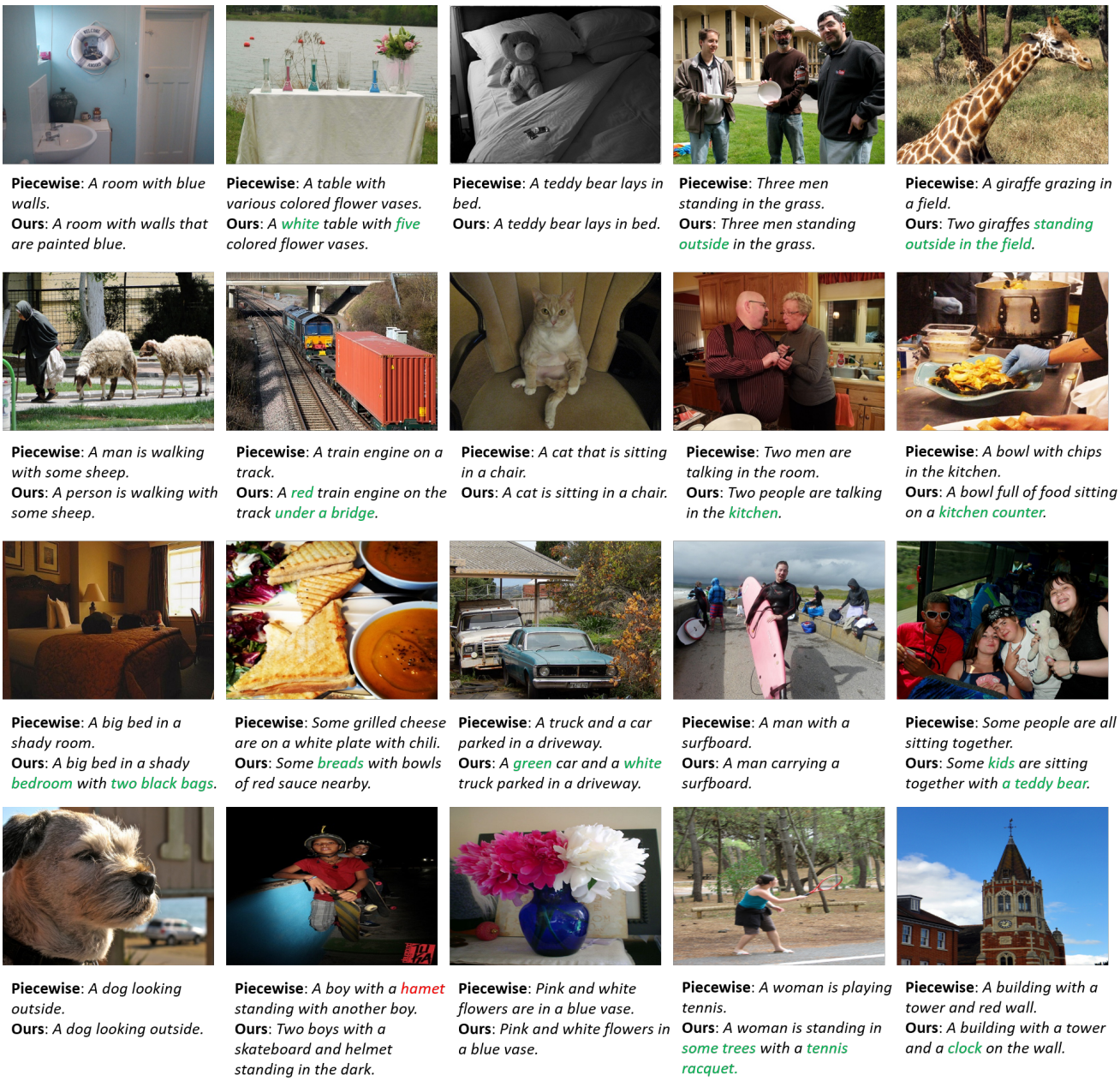

Figure 1. Image-to-speech synthesis results. Green: Fine-grained and correct instances synthesized by our model. Red: incorrect pronunciation synthesized by the piecewise model. Audio samples are available at https://bit.ly/2U7741S

2. Skip-Modal Synthesis Results

Figure 1 shows additional image-to-speech synthesis results; we manually transcribed the synthesized audio outputs for the purpose of presentation. Consistent with the qualitative results reported in the main paper (Figure 5), we see that our approach produces more detailed descriptions and has a larger vocabulary than the baseline. We encourage the readers to visit our anonymized website and listen to the audio samples: https://bit.ly/2U7741S

3. Cross-Modal Retrieval Results

Besides the synthesis tasks, another way to evaluate the performance of our model is via cross-modal retrieval. In this section, we show qualitative results of cross-modal retrieval where we use an instance from either dataset and find the most

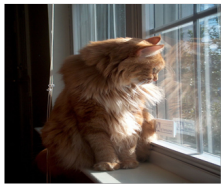

Figure 2. Cross-Modal retrieval results. The first column shows queries from each modality. The second and third columns show the top-3 retrieval results from the other two modalities. Audio samples are available at https://bit.ly/2U7741S

similar instances from different modalities from both datasets. Specifically, we compute $\mathbf{u}^j$ from all instances in the test splits of both datasets, and compute the cosine similarity between any pair of cross-modal instances.

Figure 2 shows the top 3 retrieved results in different combinations of modalities. We can see that the retrieved results are very related to the query at the object level, e.g., "dog" and "zebra" in the first and the second rows, while on the other two rows the results are related to the query at the scene/context level, e.g., "baseball game" and "birthday party". It is particularly interesting to see that the results are reasonable even for the cross-dataset retrieval settings (using an image from COCO to retrieve audio/speech instances from EMT-4). This suggests the representations extracted using our model are not very sensitive to the dataset and the modalities involved.

Batch Sampling Strategy	B@1 (I2T)	WER (S2T)	WER (T2S)
Alternative	74.1	3.88	10.5
Mixing	74.5	3.76	10.5

Table 1. Evaluation of different batch sampling strategies. I2T: image-to-text, S2T: speech-to-text, T2S: text-to-speech.

n_k (fix $d_k = 256$)	10	128	256
BLEU-1	62.5	74.1	73.9
d_k (fix $n_k = 128$)	64	128	256
BLEU-1	49.3	70.2	74.1

Table 2. Sensitivity of the memory fusion module. n_k : the number of basis vectors, d_k: the size of each basis vector.

4. Additional Ablation Experiments

4.1. Different Batch Sampling Strategies

As we trained our model on a combination of two datasets, there comes two ways to perform mini-batch training: one that samples instances from only one dataset and alternates between the two (alternate); and another that always samples instances from both datasets (mixing). We compare these two batch sampling strategies in this section. Specifically, in the first setting (alternative) we sample eight instances from either the COCO or EMT-4 dataset, while in the second setting (mixing) we sample four instances from COCO and the other four from EMT-4. We evaluate this on image-to-text (I2T), speech-to-text (S2T), and text-to-speech (T2S) synthesis tasks, reporting BLEU-1 for I2T and the word error rate (WER) for the other two.

Table 1 shows that the performance improves when we use the mixed batch sampling strategy. The improvement is especially pronounced for the text-sensitive tasks; on image-to-text synthesis the BLEU-1 is improved from 74.1 to 74.5, and on speech-to-text synthesis the WER is reduced from 3.88 to 3.76. We did not find significant differences in the text-to-speech synthesis task.

4.2. Sensitivity Analysis of Memory Fusion Module with parameters n_k and d_k

As we showed in Table 4 (ablation results) in the main paper, the memory fusion module plays an important role in our model; the performance drops most significantly when we bypass this module. It extracts compact, modality-agnostic representations of the multimodal inputs following the information bottleneck principle [1], using the shared external memory M to "bottleneck" any redundant and modality-specific information from leaking into the output representation. To better understand the behavior of this module, we analyze the sensitivity of the module to two hyper-parameters: the number of basis vectors (n_k) and the size of each basis vector (d_k) inside the external memory variable M. We evaluate this on image-to-text generation (i.e., image captioning) and report the results using BLEU-1 as our metric.

Table 2 shows our model is more sensitive to the dimension of each basic vector d_k than the number of basis vectors n_k; it achieves a significantly lower performance with $d_k = 64$ compared to any other combination of the two parameter values. The performance improves as we increase d_k, achieving the best performance when $d_k = 256$; we did not evaluate beyond $d_k = 256$ due to the limitations on the GPU memory. As for the number of basis vectors n_k, we can see the performance is low when there are only a few of them ($n_k = 10$). This shows we need a large number of basis vectors to capture the variety of information contained in multimodal data. We found that the performance is relatively stable when n_k is greater than 128.

References

[1] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. In *Allerton Conference on Communication, Control, and Computing*, 1999.

[2] Yuxuan Wang, RJ Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, Zhifeng Chen, Samy Bengio, et al. Tacotron: Towards end-to-end speech synthesis. In *INTERSPEECH*, 2017.