Supplementary material for
“CollaGAN: Collaborative GAN for Missing Image Data Imputation”

Dongwook Lee[1], Junyoung Kim[1], Won-Jin Moon[2], Jong Chul Ye[1]
[1]: Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea
{dongwook.lee, junyoung.kim, jong.ye}@kaist.ac.kr
[2]: Konkuk University Medical Center, Seoul, Korea
mdmoonwj@kuh.ac.kr

1. Collaborative training

This section describes the experiments that further analyze the importance of multi-inputs by providing additional qualitative results.

1.1. Effects of input dropout

The input of the proposed method is much more informative than StarGAN [1]. In other words, there might exist some wasted inputs since there is some redundancy of the inputs. For example on RaFD [4], if ‘Happy’ image plays a major role for reconstructing ‘Angry’ images, the other facial expressions may contribute little on the output, which is not collaborative. To achieve the collaborative learning, it is imporatant to use random nulling on the inputs (control of the number of missing imputs). Thus, the random nulling of the input images helps to increase the contribution of the other facial expressions evenly. It could be treated as a dropout [7] layer on the input images. The contribution of the input dropout is as shown in Fig. 1. The input dropout increases the performance of the reconstruction quality for all ‘Missing N’ since the inputs contribute more evenly to the reconstruction.

1.2. Incompelete input datasets

To investigate the effects of the number of inputs, we compared the reconstruction results with the control of the missing number of inputs. Figure. 2 shows the reconstruction results ‘Happy’ and ‘Angry’ using the inputs with different missing values from seven to one. As the amount of input information increased, the reconstruction results improved qualitatively as shown in Fig. 2.

2. Implementation Details

2.1. Details of MR acquisition parameter

Among the four different contrasts, three of them were synthetically generated from the MAGiC sequence (T1F, T2w and T2F) and the other was additionally scanned by conventional T2-FLAIR sequence (T2F*). The MR acquisition parameters are shown in Table. 1.

	TR(ms)	TE(ms)	TI(ms)	FA(deg)
T1F	2500	10	1050	90
T2w	3500	128	-	90
T2F	9000	95	2408	90
T2F*	9000	93	2471	160
Common parameters	FOV:220×220mm, 320×224 matrix, 4.0 mm thickness			

Table 1: MR acquisition parameters for each contrast. T1F, T2w, T2F and T2F* represent MAGiC synthetic T1-FLAIR, MAGiC synthetic T2-weighted, MAGiC synthetic T2-FLAIR and conventional T2-FLAIR, respectively. Four contrasts share the field of view (FOV), acquisition matrix, and slice thickness as shown in the common parameters row.

2.2. Network Implementation

The proposed method consists of two networks, the generator and the discriminator. There are three tasks (MR contrasts imputation, illumination imputation and facial expression imputation) and each task has its own property. Therefore, we redesigned the generators and discriminator for each tasks to achieve the best performance for each task while the general network architecture are similar.

MR contrast translation Instead of using single convolution, the generator uses two convolution branches with 1x1 and 3x3 filters to handle the multi-scale feature information. The two branches of the convolutions are concatenated similar to the inception network [8]. We called this series of two convolution, concatenation, instance normalization [9] and leaky-ReLU [2], CCNR unit, as shown in Table. 2. These CCNR units help the pixel-by-pixel processing of the CNN as well as the processing with a large FOV. The architecture of the generator describes in Table. 2 and Fig. 3.

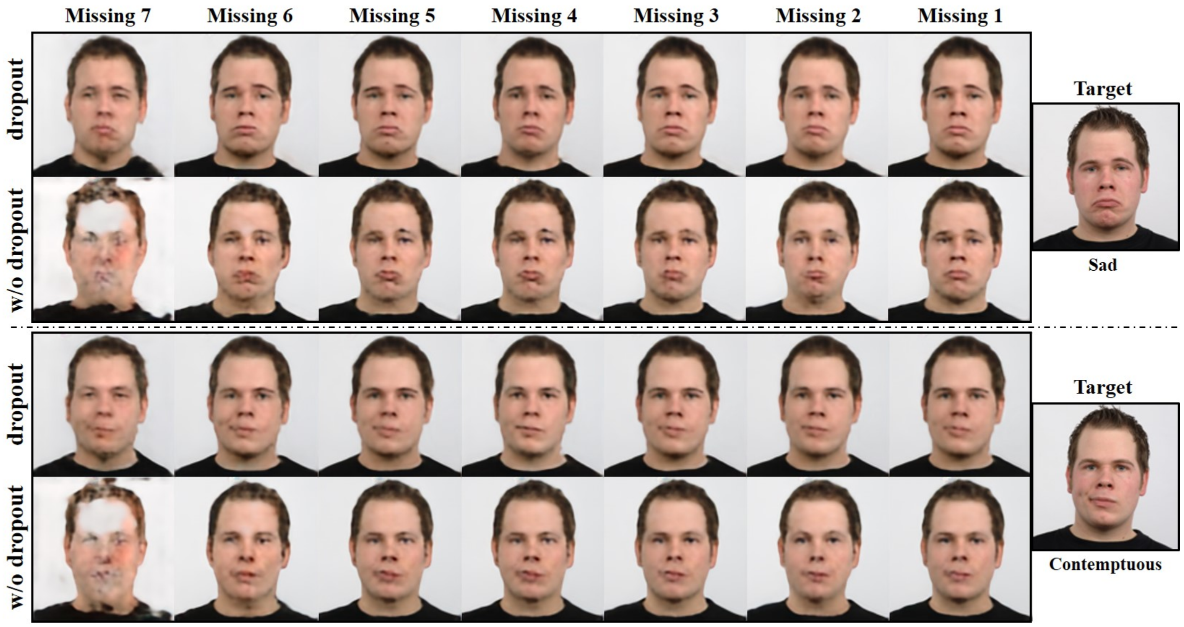

Figure 1: Facial expression imputation results changing the missing number of input images from seven to one. ‘Sad’ (up) and ‘Contemptuous’ (down) facial expressions were reconstructed using various number of inputs. Each 1st row was the results trained by input dropout and the other was not. Each column represents the results from the incomplete input set which has ‘Missing N’ inputs. To impute each facial expression, other (8-N) facial expressions were collaboratively used as inputs.

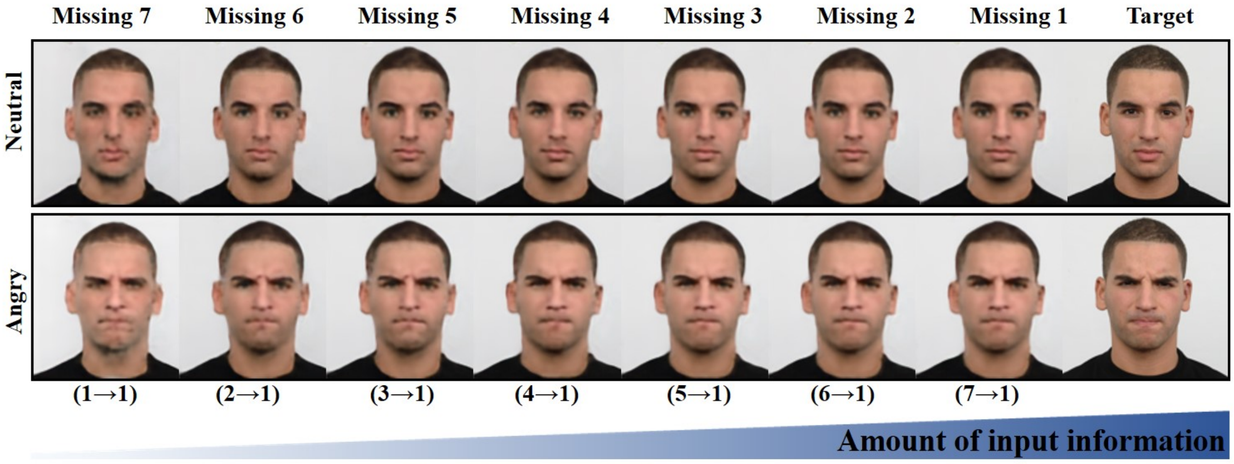

Figure 2: Facial expression imputation results changing the missing number of input images from seven to one. ‘Neutral’ (1st row) and ‘Angry’ (2nd row) facial expressions were reconstructed using various number of inputs. Each column represents the results from the incomplete input set which has ‘Missing N’ inputs. To impute each facial expression, other (8-N) facial expressions were collaboratively used as inputs. More information was used to the right column and the quality of the reconstruction improved. The numbers below the images, ($N_{in} \rightarrow N_{out}$), explain the number of input images and output images, respectively.

Unit	Layers						nCh
Main		CCNL×2	$\frac{\text{(skip)}}{\text{(Blck\#1)}}$	Cat	CCNL×2	C’	16
Blck#1	P	CCNL×2	$\frac{\text{(skip)}}{\text{(Blck\#2)}}$	Cat	CCNL×2	T	32
Blck#2	P	CCNL×2	$\frac{\text{(skip)}}{\text{(Blck\#3)}}$	Cat	CCNL×2	T	64
Blck#3	P	CCNL×2	$\frac{\text{(skip)}}{\text{(Blck\#4)}}$	Cat	CCNL×2	T	128
Blck#4	P		CCNL×2			T	256
CCNL	$\frac{\text{Conv(k1,s1)}}{\text{Conv(k3,s1)}}$	Cat-InstanceNorm-LeakyReLU					

Table 2: Architecture of the generator used for MR contrast translation. The U-net [6] structure was redesigned with the proposed CCNR units which includes instance normalization (N) and leaky-ReLU (L). Conv, P, Cat and T represent convolution, average pooling with strides 2, concatenate, and convolution transpose with strides 2 and kernel size 2×2, respectively. While k and s refer to the kernel size and the stride, C’ is 1×1 convolution layer, Conv(k1,s1).

To classify the MR contrast, multi-scale (multi-resolution) processing is important. The discriminator has three branches that each has different scales as shown in Table. 3. A branch handles the feature on the original resolution. Another branch process the features on the quater-resolution scales ($height/4, width/4$). The other one sequentially reduces the scales for extract features. Three branches are concatenated to process multi-scale features. Similar architecture with this kind of multi-scale approach works well to classify the MR contrast [5].

Order	Layers					k
1a	C(n4,s1)-L	C(n4,s1)-L	C(n4,s1)-L	C(n4,s1)-L	C(n16,s4)-L	4
1b	C(n4,s1)-L	C(n8,s2)-L	C(n8,s1)-L	C(n16,s2)-L	C(n16,s1)-L	4
1c	C(n16,s4)-L	C(n16,s1)-L	C(n16,s1)-L	C(n16,s1)-L	C(n16,s1)-L	4
2	1a 1b 1c	Cat	C(n32,s2)-L	C(n64,s2)-L	C(n128,s2)-L	4
3a	C(n1,s1)	Sigmoid (D_{gan})				3
3b	FC(n4)	Softmax (D_{cls})				8

Table 3: Architecture of the descriminator used for MR contrast translation. k is the kernel size for the convolution and C(n,s) represents the convolution layer with n channels and s strides. Cat, L and FC represent the concatenate layer, the leaky-ReLU layer and the fully-connected layer, respectively.

Illumination translation Architecture of the generator used for illumination translation. It is similar to original U-net structure with instance normalization (N) and leaky-ReLU (L) instead of batch normalization and ReLU, respectively, as shown in Table. 4 and Fig. 4.

Unit	Layers						nCh
Main	-	CNL×2	$\frac{\text{(skip)}}{\text{(Blck\#1)}}$	Cat	CNL×2	C’	64
Blck#1	P	CNL×2	$\frac{\text{(skip)}}{\text{(Blck\#2)}}$	Cat	CNL×2	T	128
Blck#2	P	CNL×2	$\frac{\text{(skip)}}{\text{(Blck\#3)}}$	Cat	CNL×2	T	256
Blck#3	P	CNL×2	$\frac{\text{(skip)}}{\text{(Blck\#4)}}$	Cat	CNL×2	T	512
Blck#4	P		CNL×2			T	1024
CNL		Conv(k3,s1)	Cat-InstanceNorm-LeakyReLU				

Table 4: Architecture of the generator used for illumination translation. Conv, P, Cat and T represent convolution, average pooling with strides 2, concatenate, and convolution transpose with strides 2 and kernel size 2×2, respectively. k and s refer to the kernel size and the stride. C’ is 1×1 convolution layer, Conv(k1,s1).

The discriminator is consists of convolutions with strides 2 and instance normalization. At the end of the discriminator, there are two branch [1]: one for discriminating real/fake and the other for the domain classification. Here, patchGAN [3, 10] was utilized to classify the source (real/fake).

Order	Layers
1	C(n64,k4,s2)-L
2	C(n128,k4,s2)-L
3	C(n256,k4,s2)-L
4	C(n512,k4,s2)-L
5	C(n1024,k4,s2)-L
6	C(n2048,k4,s2)-L
7a	C(n1,k3,s1)-Sigmoid (D_{gan})
7b	FC(n5)-Softmax (D_{cls})

Table 5: Architecture of the generator used for facial expression translation.

Facial expression translation For the generator of facial expression translation, we designed a multi-branched U-net which has individual encoder for each input images (Fig. 5). The default architecture is based on U-net structure. The generator consists of two part: encoder and decoder. In the encoding step, each image are encoded separately by eight branches. Here, the mask vector is concatenated to every input images to extract the feature for the target domain. Then, the encoded features are concatenated in the decoder and the decoder shares the structure of the modified U-net as explained in Table. 4. The discriminator shares the architecture with the one used for the illumination translation task (Table. 5) except fot the last fully-connected layer has eight channels for eigth facial expression classification.

3. Additional evaluation results

Quantitative evaluation: The results of the facial expression and illumination need to be evaluated based on the realistic image quality and the classification performance by the domain classifier. In the following, however, quantitative evaluation for facial expression and illumination imputation is provided in the form of a table in terms of NMSE (normalized mean squared error) and SSIM (structural similarity index).

Additionally, we also presented the results of pix2pix [3] which is a single-pair supervised method, to understand whether the proposed multiple cycle consistency losses actually allow for even better performance.

	pix2pix	CycleGAN	StarGAN	Proposed
A	0.0247	0.0301	0.0306	**0.0197**
	0.765	0.732	0.698	**0.794**
C	0.0283	0.0327	0.0421	**0.0105**
	0.724	0.0700	0.696	**0.840**
D	0.0333	0.0362	0.0397	**0.0172**
	0.716	0.694	0.683	**0.802**
F	0.0395	0.0329	0.0487	**0.0213**
	0.677	0.685	0.670	**0.761**
H	0.0345	0.0350	0.0420	**0.0211**
	0.697	0.682	0.606	**0.778**
S	0.0335	0.0268	0.0363	**0.0122**
	0.697	0.729	0.692	**0.803**
Sad	0.0349	0.0352	0.0395	**0.0204**
	0.679	0.6975	0.652	**0.776**

Table 6: Quantitative results for facial expression imputation. The NMSE/SSIM (lower/upper part for each facial expression, respectively) are calculated from each target domain (A:angry, C:contemptuous, D:disgusted, F:fearful, H:happy, S:surprised, Sad:sad).

	pix2pix	CycleGAN	StarGAN	Proposed
$-90°$	0.0334	0.0777	0.0545	**0.0122**
	0.799	0.640	0.606	**0.876**
$-45°$	0.0181	0.0656	0.0470	**0.00873**
	0.840	0.688	0.644	**0.888**
$45°$	0.0151	0.0188	0.0178	**0.0150**
	0.607	0.734	0.698	**0.800**
$90°$	0.0680	0.0868	0.0481	**0.00839**
	0.708	0.665	0.668	**0.894**

Table 7: Quantitative results for illumination imputation. The NMSE/SSIM (upper/lower part for each row, respectively) are calculated from the target domain.

Table 6 & 7 show the additional quantitative evaluation result showing that CollaGAN is better compared to the other algorithms. Here, pix2pix[3], which directly imposes the loss between the generator output and the target data, was also used for the comparison. While pix2pix[3] shows better reconstruction performance compared to CycleGAN and StarGAN, the proposed method shows the best performance as shown in Table 6 & 7 even for paired dataset.

Additional qualitative evaluation : We performed an additional quality assessment by Mechanical Turk experiment for more elaborate qualitative evaluation on the reconstruction results (Table. 8). We asked 30 participants to select the best image according to the image quality and how well the result represents the facial expression of the target domain. 70.8% of the reconstruction results from CollaGAN was chosen as the best reconstruction.

Chosen as the best	pix2pix	CycleGAN	StarGAN	Proposed
	3.8%	17.9%	7.4%	**70.8%**

Table 8: Qualitative evaluation results using Mechanical Turk experiment. We asked the participants to choose the best image according to the quality of reconstruction image, the similarity to the ground truth, and how well the original facial expression is expressed. Total 1470 answers from 30 participants.

4. Ablation study

To verify the advantage of the proposed multiple-cycle consistency (MCC) loss and SSIM loss, ablation studies were performed using RaFD dataset, and the results are presented in Table 9.

(Mean±std)	l_1 w/o L_{MCC}	w/o L_{SSIM}	Proposed
NMSE	0.0372±0.00653	0.0200±0.00391	**0.0178±0.00419**
SSIM	0.714±0.0211	0.779±0.0243	**0.793±0.0237**

Table 9: Quantitative results for the ablation study.

When multiple cycle consistency loss was replaced with l_1 loss (i.e. direct regression from multiple inputs to the single target), the results showed inferior performance compared to the proposed method. Also, we found that L_{SSIM} improved the reconstruction performance in terms of NMSE and SSIM (Table 9).

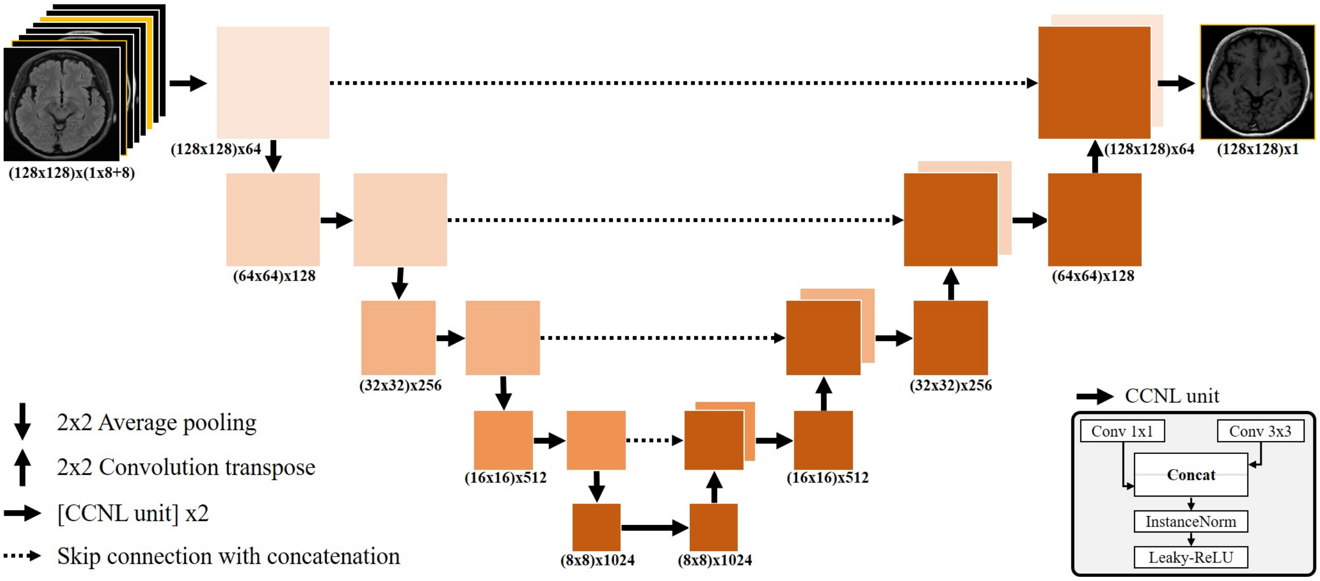

Figure 3: Architecture of the generator used for MR contrast imputation. It is the modified U-net architecture with CCBR unit which consists of two branches of convolution (3×3 and 1×1), concatenation, instance normalizatoin and leaky-ReLU. The input images were concatenated with mask vector which represents the target domain. The downward arrows, upward arrows, right arrows and dashed arrows represent 2×2 average pooling, 2×2 convolution transpose, two repetition of CCNL unit and skip connection with concatenation, respectively, as explained in Table. 2

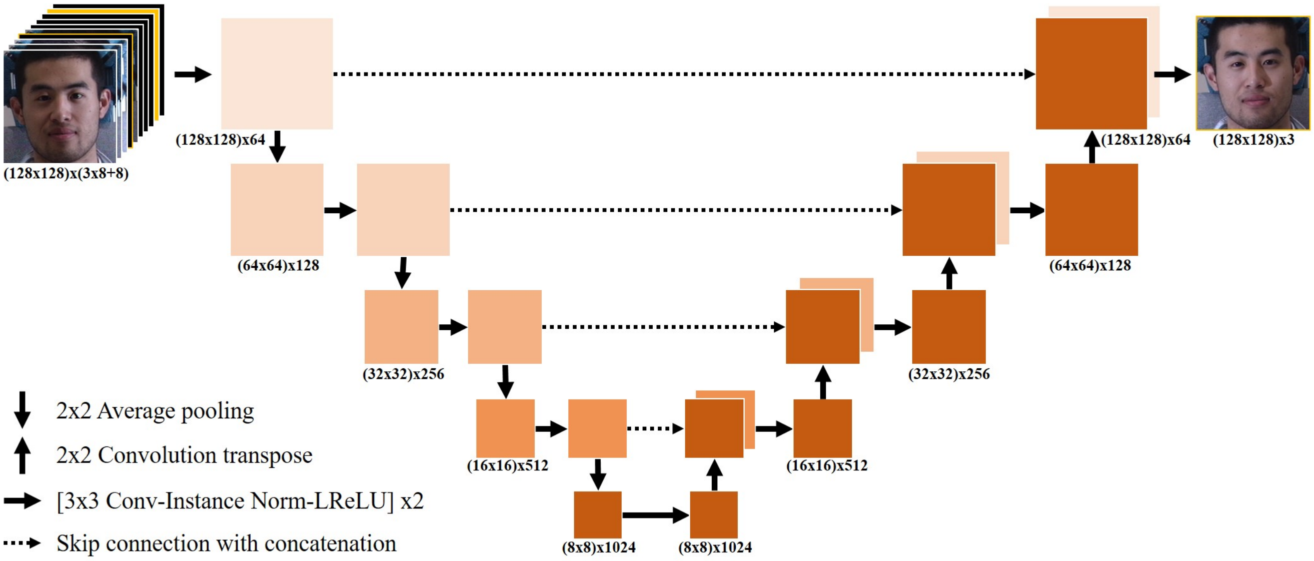

Figure 4: Architecture of the generator used for illumination imputation. U-net structure with instance normalization and leaky-ReLU was used. The input images were concatenated with mask vector which represents the target domain. The downward arrows, upward arrows, right arrows and dashed arrows represent 2×2 average pooling, 2×2 convolution transpose, two repetition of [3x3 convolution, instance normalization and leaky-ReLU], and skip connection with concatenation, respectively, as explained in Table. 4

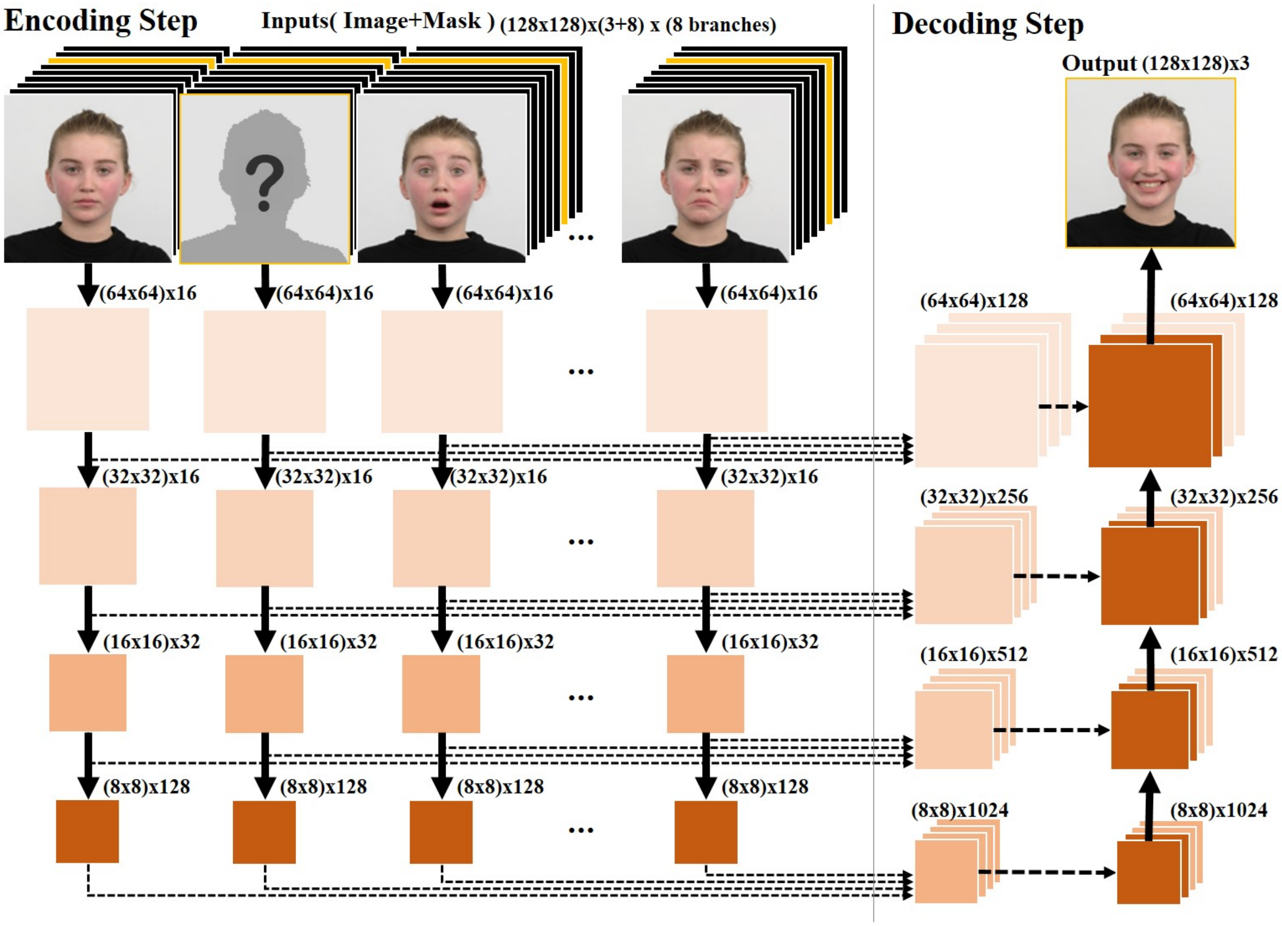

Figure 5: Architecture of the generator used for facial expression translation. It has multi-branched encoder for individual feature extraction of each input images. The encoded features are concatenated in the decoder and the decoder structure shares with the discriminator used for the illumination translation. $(h \times w)\times N_{ch}$ represents the dimension of the features/images where h, w and N_{ch} is height, width and number of channels. The dashed arrow means skip connections. The downward, upward and right arrows represent [CNL×2-P] layers, [T-CNL×2] layers and [CNL×2] layers, respectively, as explained in Table. 4

4.1. Additional Qualitative Results

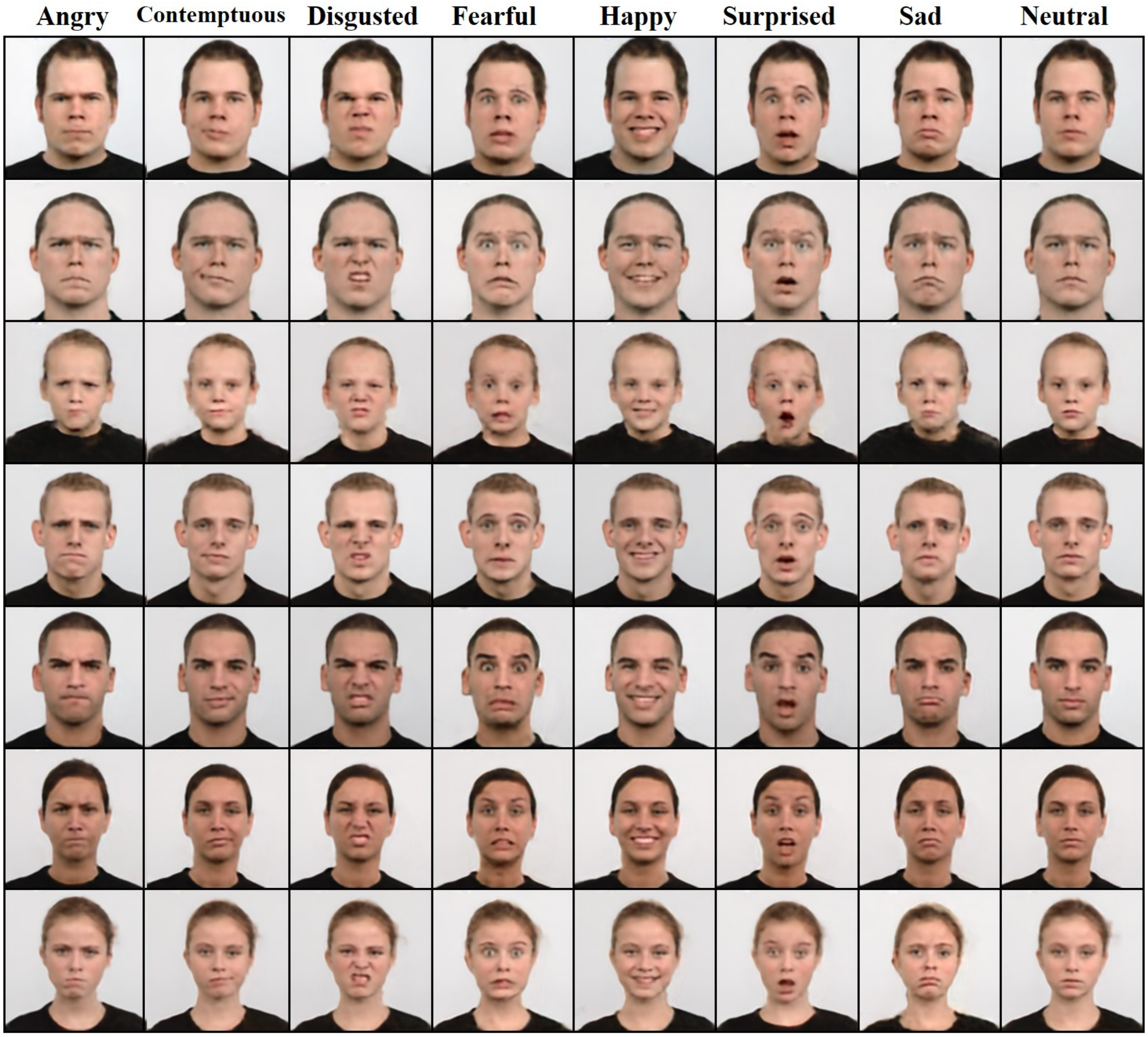

Figure 6: Additional results for facial expression imputation. To impute each facial expression, the other seven facial expressions were collaboratively used as inputs.

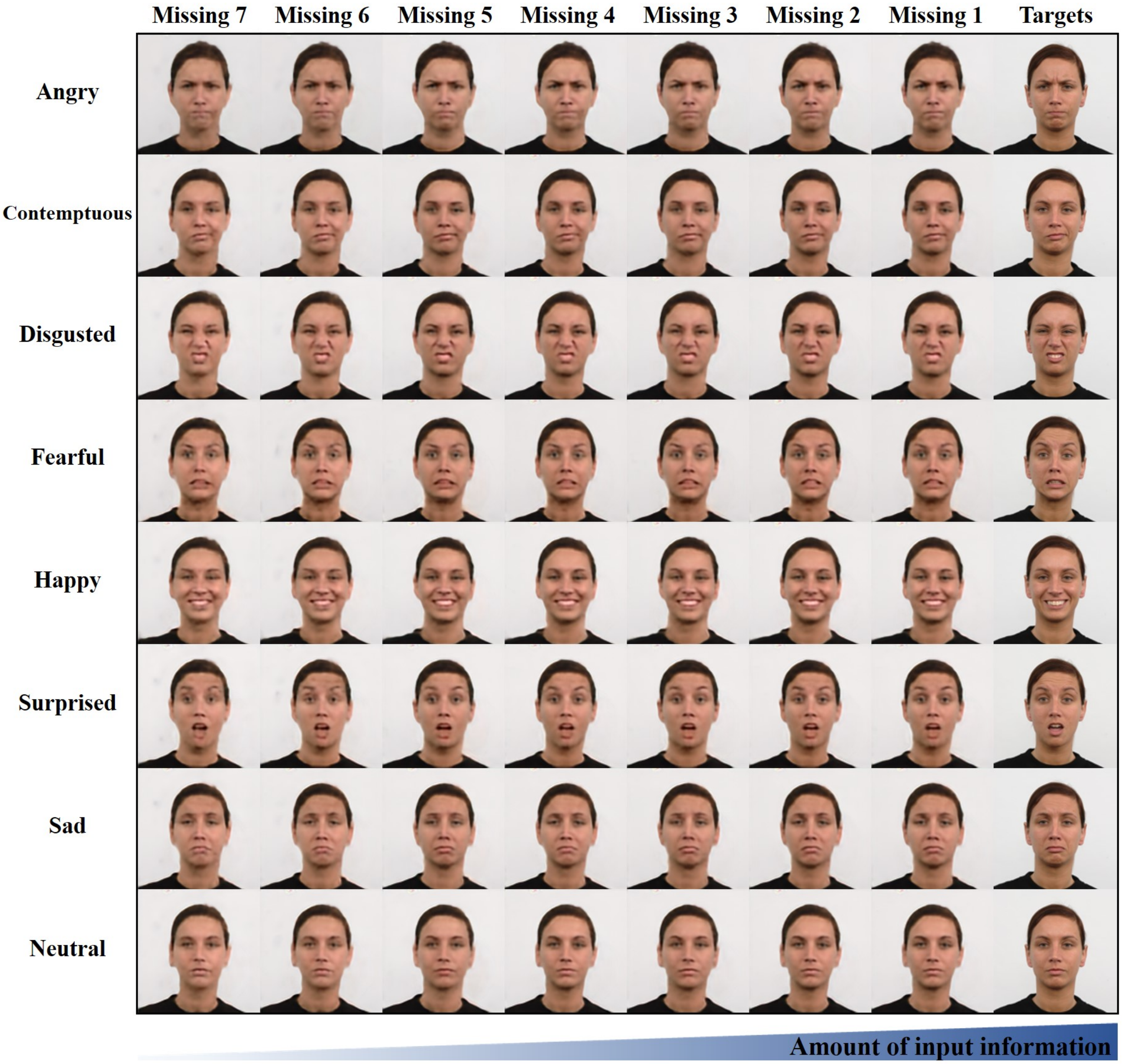

Figure 7: Additional results for facial expression imputation from incomplete input sets. Each column represents the results from the incomplete input set which has ‘Missing N’ inputs. To impute each facial expression, other (8-N) facial expressions were collaboratively used as inputs. More information was used to the right column.

References

[1] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. *arXiv preprint*, 1711, 2017.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[3] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint*, 2017.

[4] O. Langner, R. Dotsch, G. Bijlstra, D. H. Wigboldus, S. T. Hawk, and A. Van Knippenberg. Presentation and validation of the radboud faces database. *Cognition and emotion*, 24(8):1377–1388, 2010.

[5] S. Remedios, D. L. Pham, J. A. Butman, and S. Roy. Classifying magnetic resonance image modalities with convolutional neural networks. In *Medical Imaging 2018: Computer-Aided Diagnosis*, volume 10575, 2018.

[6] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[7] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[8] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[9] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[10] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint*, 2017.