A. Supporting Information Document S1 (SSIM)

In general, SSIM quality metrics is comprised of the multiplication of the three terms, including the luminance term $L(.)$, contrast term $C(.)$, and structural term $S(.)$. SSIM per pixel/voxel between two 2D/3D images A and B can be formulated as Equation 1[1].

$$SSIM(x,y) = [L(x,y)]^{\alpha}[C(x,y)]^{\beta}[S(x,y)]^{\gamma} \quad [1]$$

Where A and B are inputs to all functions, but they are omitted for the sake of clarity .The x and y are the pixel/voxel intensity values from the input images.
Luminance, contrast, and structural terms can be defined as Equations (2-4):

$$L(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \quad [2]$$

$$C(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \quad [3]$$

$$S(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \quad [4]$$

where μ_x, μ_y, σ_x, σ_y, and σ_{xy} are the local means, standard deviations, and cross-covariance. By considering $\alpha = \beta = \gamma = 1$, and $C_3 = C_2/2$, which has been proposed by Wang et al.[1], the original SSIM quality metric (Eq. 1) can be simplified to Equation 5.

$$SSIM(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \times \frac{2\sigma_{xy} + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \quad [5]$$

where C_1 and C_2 are two small constants to stabilize the division with a weak denominator, local statistics are computed by applying the 2D/3D Gaussian filter with standard deviation σ_G.
Finally, the mean of the calculated SSIM map, a scalar value, can be used as a similarity metric between two given 2D/3D images. Equation 6 shows the mean of the SSIM map, which commonly used as an image quality assessment metric or loss function in the neural networks:

$$MSSIM(A,B) = \frac{1}{N}\sum_x \sum_y SSIM(x,y) \quad [6]$$

N is the number of the pixels/voxels inside the input images.
In our work, we used SSIM as the part of the loss function in the 3D U-Net and the generator part of the GANs. To calculate the SSIM for the 2D GAN, C_1 =0.0001 and C_2 =0.0009, 2D Gaussian filter with window size = 11×11, and σ_G=1.5 were used. To calculate the SSIM for the 3D networks including TAV-GAN, Temporal-GAN, Volumetric-GAN, and 3D U-Net, C_1 =0.0001 and C_2 =0.0009, 3D Gaussian filter with window size = 11×11×11, and σ_G=1.5 were used.

B. Supporting Information Document S2 (2D GAN)

B1. Network architecture. The detailed network architecture for 2D GAN is shown in Figure SB1. The generator is a 2D U-Net which consists of two paths: (I) the encoder path, which includes four downsampling blocks; (II) the decoder path, which contains four up-sampling blocks. Each block has two convolutional layers, with each layer containing learnable convolution filters followed by the non-linear activation function Leaky ReLU (LReLU). Convolutional layers in the first block of the network contain 64 convolutional kernels, and the number of kernels doubles in each deeper block. Down-sampling and up-sampling blocks in the encoder and decoder paths are connected via average pooling (strides = 2) and up-sampling (strides = 2). A skip connection is used to pass the data between each pair of same-sized up-sampling and down-sampling blocks. The discriminator is a 2D binary classifier which contains four downsampling blocks. Each block contains two convolutional layers in which each convolutional layer contains convolutional kernels followed by LReLU. The starting number of channels used in the discriminator was 64, which was doubled in each deeper block. The last two layers are the fully connected layer followed by dropout and LReLU, and a single decision fully connected layer with a sigmoid activation function. Discriminator takes the magnitude of the generated images to decide whether it is “generated” or “clean” images. The input and output of the generator for the 2D GAN in the training stage is a complexed-valued image patch with size 320×192×2 (real and imaginary), and magnitude-valued image patch with size 320×192×1, respectively.

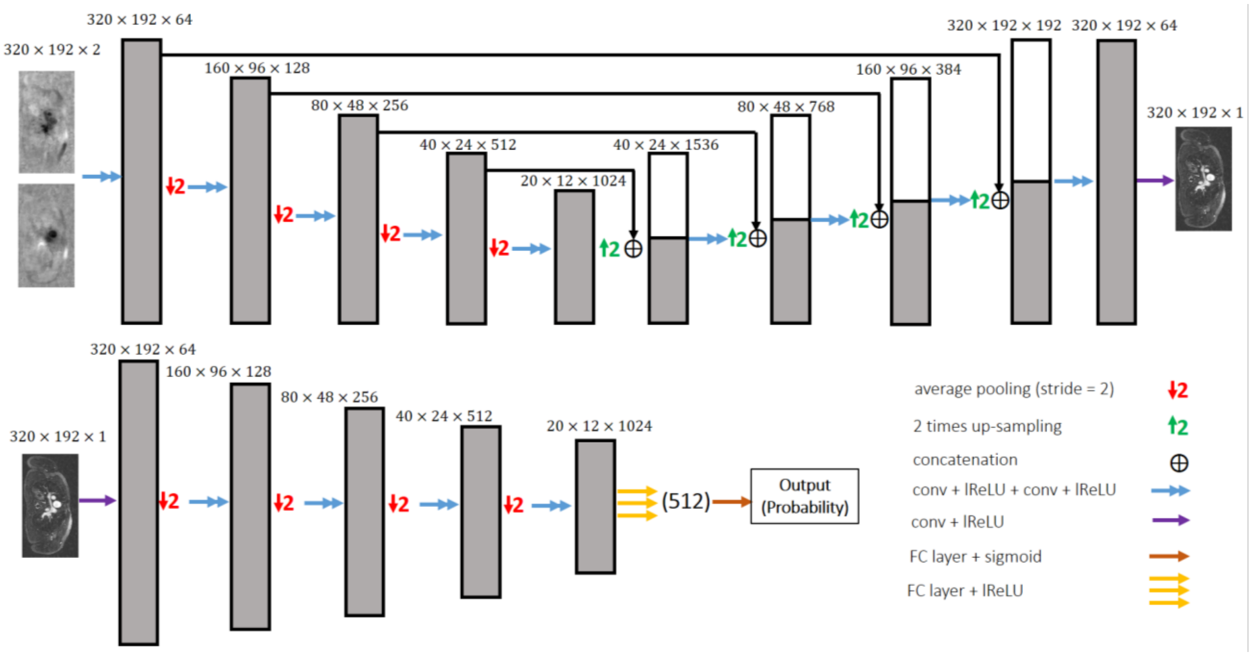


Figure SB1. The detailed network structure for 2D GAN. The generator part is a 2D U-Net with 4 downsampling blocks and 4 up-sampling blocks. The discriminator part is a 2D binary classifier with four downsampling blocks. The number of the convolutional kernels and type of the activation functions are reported in the Figure. Network training was performed on the image patches with size 320×192.

B2. Loss function.

The total loss function of the generator part of the 2D GAN $L_{G^{2D}}^{Total}(.)$ is a linear combination of the adversarial loss $L_{G^{2D}}^{a}(.)$, normalized L$_1$ norm, and SSIM$_{2D}$. The total loss function of the discriminator $L_{D^{2D}}^{Total}(.)$ is an adversarial loss $L_{D^{2D}}^{a}(.)$. Equations (1, 2) formulated the generator’s objective function and the discriminator’s objective function, respectively:

$$\min_{\theta_{g^{2D}}} L_{G^{2D}}^{Total}\left(x^{i,t},\, G^{2D}(\tilde{x}_u^{i,t};\theta_{g^{2D}})\right) = \min_{\theta_{g^{2D}}} \gamma\left[L_{G^{2D}}^{a}\left(D^{2D}(x^{i,t};\theta_{d^{2D}}),\, G^{2D}(\tilde{x}_u^{i,t};\theta_{g^{2D}})\right)\right] + \lambda\left[\frac{1}{N}\left\|x^{i,t} - G^{2D}(\tilde{x}_u^{i,t};\theta_{g^{2D}})\right\|_1\right] - \zeta\left[SSIM_{2D}\left(x^{i,t}, G^{v}(\tilde{x}_u^{i,t};\theta_{g^{2D}})\right)\right] \quad [1]$$

$$\min_{\theta_{d^{2D}}} L_{D^{2D}}^{Total}\left(D^{2D}(x^{i,t};\theta_{d^{2D}}),\, G^{2D}(\tilde{x}_u^{i,t};\theta_{g^{2D}})\right) = \min_{\theta_{d^{2D}}} \gamma\left[L_{D^{2D}}^{a}\left(D^{2D}(x^{i,t};\theta_{d^{2D}}),\, G^{2D}(\tilde{x}_u^{i,t};\theta_{g^{2D}})\right)\right] \quad [2]$$

Where $\tilde{x}_u^{i,t}$, $x^{i,t}$ stands for the aliased and respiratory motion-corrupted, and un-aliased and free of the motion 2D image patches for the t^{th} cardiac phase of the i^{th} patient case. γ , λ, and ζ are the hyperparameters that control the contribution of the adversarial loss, spatial sparsity and local patch wise similarity. N is the normalization factor and is equal to the number of the pixels inside $x^{i,t}$.

B3. Training procedure. The training process for the 2D GAN is similar to the training process of the TAV-GAN. As shown in Figure SB2, the training process consists of five stable phases and four transition phases. The training started with a first stable phase. Only the layers with the lowest resolution are built and trained for an epoch in the first stable phase. Then first transition phase is started where new layers are added gradually to the lowest resolution layer to transit to the second stable phase. It is important to emphasize that as shown in Figure SB2, after each transition resolution of the image is doubled. In the transition phase, new layers were added with weight 1-α to the existing layers with weight α. The parameter α was linearly decreased from 1 to 0 through the iterations of the epoch's number. For instance, from the beginning of the transition phase (α=1), the newly added layers were getting zero weight, and as α decreases, the new layers had more weight until the part of the existing layers were faded (α=0). Once α reached 0, the transition phase was finished, and the next stable phase was started. These stable and transition phases were alternated while more layers were added progressively until the stable phase 5 was finished, which concluded the training process. Figure SB3 shows the first stable and transition phases for the 2D GAN. For the first to fourth stable and transition phases, the network is trained for an epoch. The number of the required epochs for the last stable phase is decided empirically. Two criteria for stopping the training process were considered: 1) outputs' quality through the training and 2) equilibrium state of the adversarial loss for the generator and the discriminator.

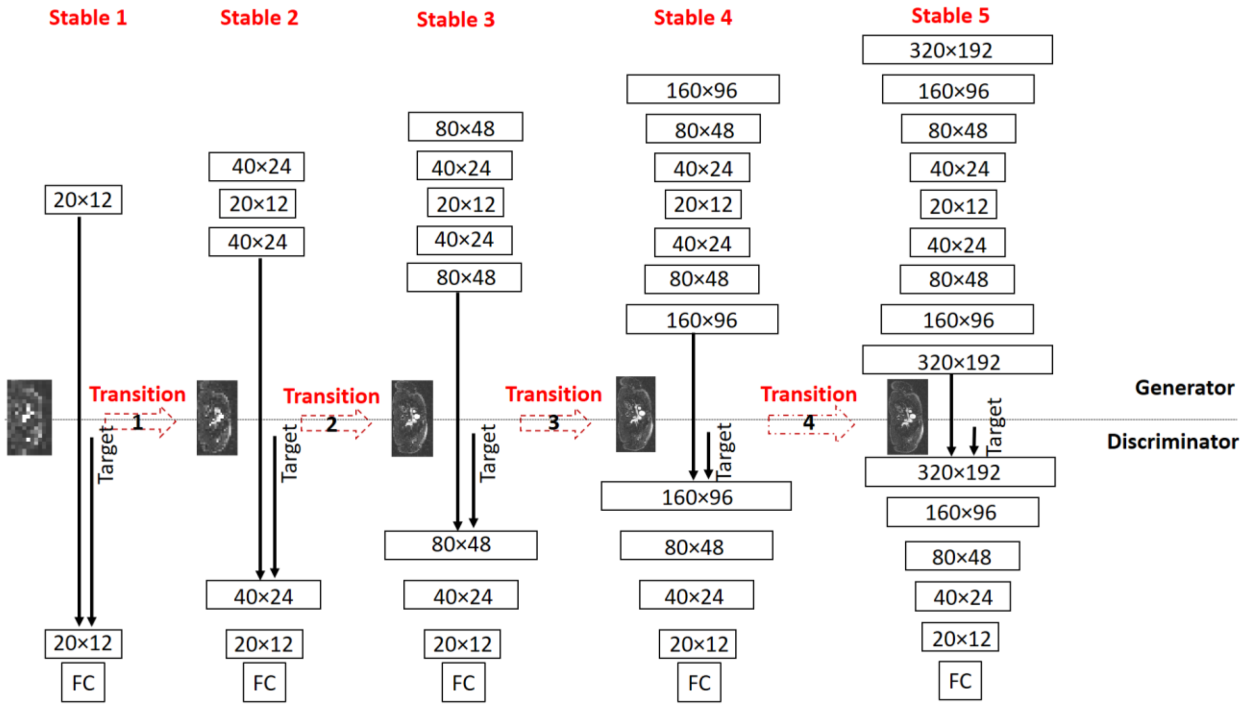


Figure SB2. Progressive training strategy for 2D GAN. Intuitively, building the network with few layers with low resolution and training them and gradually adding more layers to reach the high-resolution images can alleviate the training process of the GANs. The training procedure contains five stable phases and four transition phases. As can be seen, in the stable phase 1, only layers with the lowest resolution were built. In the transition phase 1, new layers were gradually added to the old layers to reach stable phase 2. Parameter α controls the rate of gradual pointwise addition. It linearly reduced from 1 to 0 through the iterations of the training in each transition phase. Sample of transition and stable phases were explained in Figure SB3. This alternation between stable and transition phases was continued until to reach to the last stable phase 5.

For the last stable phase, training was performed for the number of epochs. The number of the required epochs was decided based on the quality of the test results in the training stage, and the equilibrium state of the generator loss and the discriminator loss.

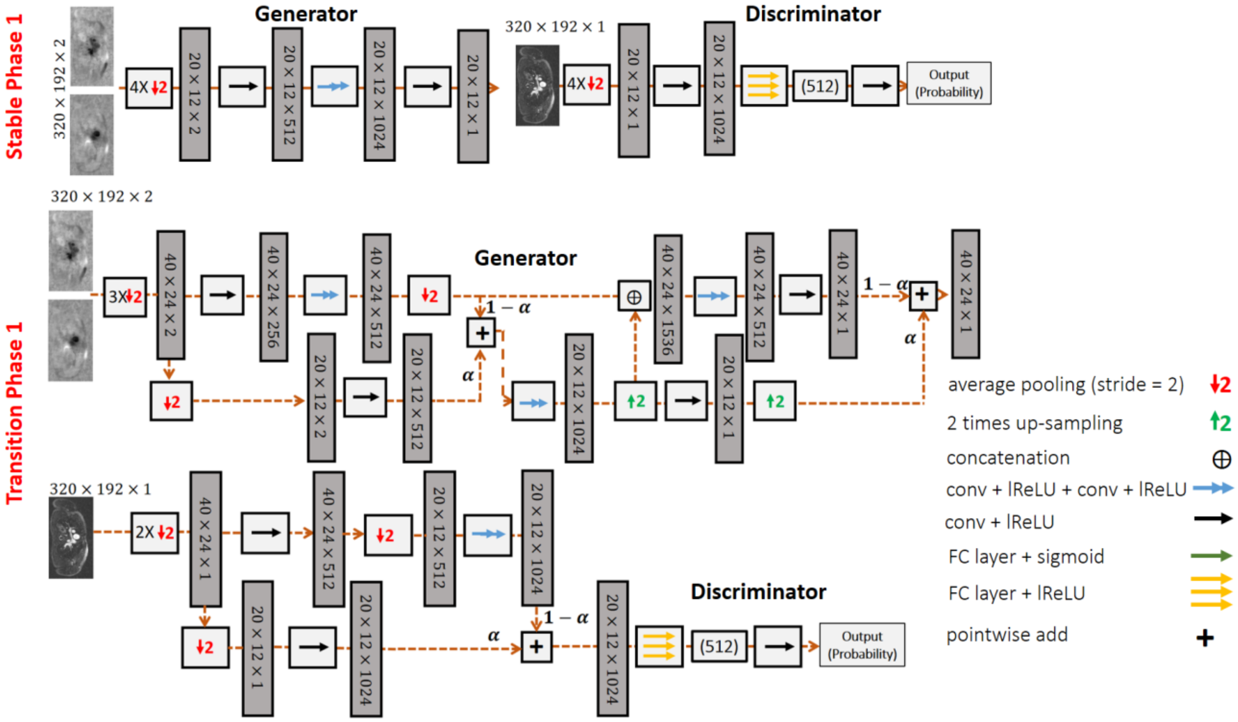


Figure SB3. Illustration of the stable and transition phases of the 2D GAN in this work. For the sake of simplicity, we only showed the first stable and transition phases. Only layers with the lowest resolution were built for the generator and the discriminator in the first stable phase. The input complex image was downsampled four times and fed to the generator. The first convolution layer in the generator and the discriminator is increasing the channel dimensions of the input. The network was trained for an epoch in the first stable phase. Then, in the first transition phase, layers with twice resolutions were added gradually to the pre-trained layers. As can be seen, new layers were added to the generator and the discriminator progressively. The parameter α controls the addition process. It is linearly decreasing from 1 to 0 through all iterations in the epochs. We trained this phase only for an epoch. To make the idea clear, for α=1, we are at the beginning of the transition phase. For α=1, the graph for the generator and the discriminator is the same as the graph in the stable phase 1. Suppose α=0; it means that the first transition phase is finished, and training will enter the second stable phase. By considering α=0, it can be seen that adapting layers in the first stable phase were faded, and new layers with higher resolution were added to the graph.

B4. Training parameters.

For the 2D GAN, $\gamma = 1,\ \lambda = 0.6$ and $\zeta = 0.4$ are selected based on the limited search as the weight of the adversarial loss, normalized L$_1$-loss, and SSIM$_{2D}$ loss. Adam optimizer was used with the momentum parameter β =0.9, mini-batch size= 64, an initial learning rate of 0.0005 for the generator, and an initial learning rate 0.00005 for the discriminator. Weights of the network are initiated with random normal distributions with a variance of σ = 0.01 and mean μ=0. The training was performed with the Pytorch interface on a commercially available graphics processing unit (GPU) (NVIDIA Titan RTX, 24GB RAM).

C. Supporting Information Document S3 (Data-Preparation)

As shown in Figure SC1 (b), each ROCK MUSIC[2] scan continuously acquired N_L Cartesian k-space lines grouped in quasi-spiral interleaves in the Ky-Kz plane that are arranged in a golden-angle manner, shown in Fig. SC1 (a). For each ROCK MUSIC raw data in Group A, a pair of image volumes were reconstructed for network training: the reference image and the highly accelerated, aliased and respiratory motion-corrupted image. To reconstruct the reference image, data were binned into 9-12 cardiac phases of the end-expiration respiratory state by using the cardiac and respiratory self-gating signal derived from the k-space center lines as shown in Figure SC1(b) and reconstructed based on Equation 1[2,3]:

$$\hat{d} = \underset{d}{\operatorname{argmin}} \sum_{i=1}^{N} \|DFS_i d - m_i\|_2^2 + \lambda_1 \|R_1 d\|_1 + \lambda_2 \|R_2 d\|_1 \qquad [1]$$

Where F, S_i, and D are the Fourier transform, sensitivity maps, and undersampling mask, respectively. d is the multiphase images, m_i is the acquired undersampled k-space from each of the N receiver coil channels. R_1 is the spatial wavelets and R_2 is the temporal total variation. Hyperparameters λ_1 and λ_2 control the weight of the regularizers R1 and R2, respectively. The k-space under-sampling factor after cardiac and before respiratory motion SG ranged 2.8X-7.9X. To reconstruct the "highly accelerated" image volume, as shown in Fig. SC1 (c), we extracted the first $M = min\ (50000, N_L/2)$ k-space lines out of the data, resulting in a further retrospective under-sampling of the acquired data by a factor of at least 2. Because the quasi-spiral k-space interleaves were arranged in a golden-angle manner, the k-space sample uniformity is maintained even when the second half (or more) of acquired data was discarded. The total k-space under-sampling factor was 10.7X-15.8X for the "highly accelerated" image volumes.

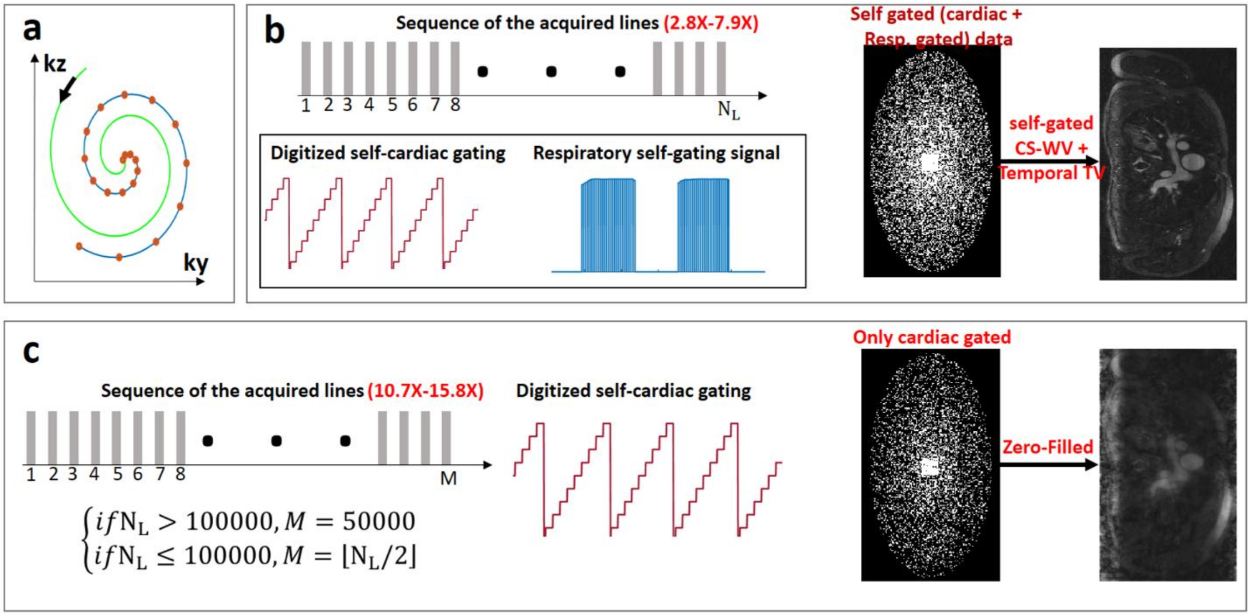

a
kz
ky
b
Sequence of the acquired lines (2.8X-7.9X)
1 2 3 4 5 6 7 8
N_L
Digitized self-cardiac gating
Respiratory self-gating signal
Self gated (cardiac + Resp. gated) data
self-gated CS-WV + Temporal TV
c
Sequence of the acquired lines (10.7X-15.8X)
1 2 3 4 5 6 7 8
M
$\begin{cases} if\, N_L > 100000, M = 50000 \\ if\, N_L \leq 100000, M = \lfloor N_L/2 \rceil \end{cases}$
Digitized self-cardiac gating
Only cardiac gated
Zero-Filled

Figure SC1. Data preparation process: (a) shows the ROtating Cartesian K-space (ROCK) sampling strategy used to acquire the data. (b) shows the SG CS-WV reconstruction process to create the clean reference volumetric images. (c) shows the zero-filled reconstruction process to create the aliased, respiratory motion-corrupted images. As shown in (c), the first half of the acquired lines (if N_L<100000 lines) or the first 50000 of the acquired lines (if N_L>100000) were used to create the inputs for training and testing the network. Also, only a self-cardiac gating signal is used to sort the data to multiple cardiac phases. No respiratory motion gating was performed when generating the input images in (c).

We note that, although the data were retrospectively under-sampled, we expect our data to accurately represent a prospectively under-sampled in vivo imaging scenario with the same under-sampling factor, because the prospective data would have been acquired using the exactly the same sequence timing and temporal order for the k-space lines and quasi-spiral interleaves. We subsequently binned the resulting highly accelerated k-space data into appropriate cardiac phases using the cardiac-gating signal, zero-filled each cardiac phase data, performed an inverse Fourier transform, and finally combined the resulting multi-coil images to a single complex coil image using fast coil combination algorithm. The highly accelerated images, in the absence of any compressed sensing reconstruction and respiratory motion gating, had significant under-sampling aliasing artifacts and respiratory motion artifacts. Both the reference images and the highly accelerated images were normalized by subtracting the complex mean within the image volume and dividing by the absolute value of twice the standard deviation of the same volume. The highly accelerated image volumes were formatted as individual 4D tensors with its complex values expressed as real and imaginary channels. The magnitude of the normalized reference images were formatted as a 4D tensor as well with a single (magnitude) channel. To minimize the background effect, 10 voxels from the edge of the tensors were cropped. To prepare for network training, paired patches, of size 64×64×64×2 from the highly accelerated images and of size 64×64×64×1 from the reference images, were extracted randomly from the cropped tensors and used as an input and target, respectively, in the training phase of the 3D U-Net, the Volumetric-GAN, and the TAV-GAN. For training the Temporal-GAN, the input was formatted as a real-valued 4D tensor with the magnitude of the three sequential cardiac phases t-1, t, t+1 in the channel dimension of the tensor, and the training target was the magnitude of the reference image corresponding to cardiac phase t. Subsequently, paired patches with sizes 64×64×64×3 for the input, and 64×64×64×1 for the target 4D tensor, was extracted randomly and used as an input and target in the training phase for the Temporal-GAN. It is worth noting that in the Temporal-GAN, to prepare the data for the first and last cardiac frames, cardiac frames were assumed cyclic. For instance, for the last cardiac frame t as the target, three aliased and respiratory corrupted cardiac frames t-1, t, 1 were stacked in the channel dimension as the Temporal-GAN input.

For 2D GAN, the reference images and the highly accelerated images were normalized slice-by-slice by subtracting the complex mean within the image slice, followed by division by the absolute value of the standard deviation of the slice. The input and target of the generator for the 2D GAN in the training stage was cropped from the slice-by-sliced normalized highly accelerated images and reference images and formatted as a complexed-valued tensor with size 320×192×2 (real and imaginary), and a magnitude-valued tensor with size 320×192×1 , respectively.

For the testing data sets in Groups B1 and B2, we reconstructed both the reference image volumes and the highly accelerated image volumes as well. Although data in these Groups were not used in network training, the reference images were used in the network performance evaluations and comparisons.

D. Supporting Information Document S4 (Sharpness Analysis)

The normalized Tenengrad focus measure[4,5] was used to quantify the sharpness of the reconstructed respiratory motion-corrected results with different networks. In general, to compute the Tenengrad focus measure, the image is convolved with a Sobel operator, and the square of all the magnitudes greater than a threshold is reported as a focus measure. Equation 1 formulates the Tenengrad measure:

$$F_{Tenengrad} = \sum_{i,j}[I(i,j) ** S]^2 + [I(i,j) ** S^T]^2 \ , \qquad [1]$$

where $I(i,j)$ shows the image and S is the Sobel operator: $S = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -2 \end{bmatrix}$.

Because of the difference in the size of the testing cases, the mean of the Tenengrad focus measure without threshold was calculated and reported as a sharpness score of an image. To calculate the results' sharpness score from different methods, we first cropped the cardiac region, and then we computed the mean of the Tenengrad focus measure for each slice of the cropped region and normalized them based on the calculated mean of the Tenengrad measure for the corresponding slice of the cropped region in the reference SG CS-WV images. Then, the normalized values were averaged over the slices inside a cropped cardiac region and cardiac phases to represent a single sharpness number for each case. We excluded the 2D GAN in our sharpness analysis because of its inferior image quality with more residual artifacts than other methods.

E. Supporting Information Document S5 (3D spatiotemporal GAN)

E1. Purpose: The goal of this supplemental study is to compare a 3D spatiotemporal GAN against the Temporal-GAN qualitatively and quantitatively.

E2. Method: A 3D spatiotemporal GAN was trained based on the ROCK MUSIC data in this work. To circumvent limitations in GPU memory, we first performed a Fourier Transform on the ROCK MUSIC data in the readout direction, to divide the 4D (3D spatial + cardiac phase) data into a contiguous series of 2D dynamic slices in the readout direction, each slice having two spatial dimensions and one temporal dimension. We included 9 cardiac phases for each 2D dynamic slice. The same Fourier Transform in the readout direction was performed for both the highly-accelerated motion-corrupted datasets and the SG CS-WV reference datasets. We subsequently trained a 3D spatiotemporal GAN that takes these individual 2D dynamic slices as the input such that the GPU memory is not saturated. This 3D spatiotemporal GAN takes advantage of 2D spatial information and the temporal information, i.e., redundant information through the sequential 2D cardiac frames, to recover the clean images from the aliased and respiratory motion affected images. The network structure for the 3D spatiotemporal GAN is similar to the Temporal-GAN (see Sup. Info. Fig. S1), except that the last convolutional layer of the network has nine kernels. For the 3D spatiotemporal GAN, a combination of two-loss functions including the content loss ($\lambda = 0.5$, $\zeta = 0.3$), and adversarial loss ($\gamma = 1$) were considered. The progressive training strategy as described in the main manuscript was used to train the network. The network's trainable weights were initiated with random normal distributions with a variance of $\sigma = 0.01$ and mean μ=0. For the 3D spatiotemporal GAN, the Adam optimizer was used with the momentum parameter β =0.9, mini-batch size= 16, an initial learning rate of 0.0001 for the generator, and an initial learning rate of 0.00001 for the discriminator. To evaluate the image quality of the 3D spatiotemporal GAN, we randomly selected 7 cases from Group B1 and Group B2, and asked two blinded radiologists to rank reconstructed dynamic image volumes using either the Temporal-GAN and the new 3D spatiotemporal GAN.

E3. Results: Figure SE1 shows the qualitative reconstruction and respiratory motion correction results for the two patient cases drawn from the datasets Group B1 and Group B2 for the three techniques, including the (a)Temporal-GAN, (b) 3D spatiotemporal GAN, and (c)2D GAN. The first row of each subpanel in Figure SE1 shows a coronal section of the results, and the second and third rows show the zoomed cardiac region and the temporal difference maps between two sequential cardiac frames. Based on the temporal difference maps in both patient cases, the flickering artifacts were substantially reduced in both Temporal-GAN and the 3D spatiotemporal GAN in comparison to the 2D GAN. Both Temporal-GAN and the 3D spatiotemporal GAN had better performance in removing the aliasing and respiratory artifacts from the image than the 2D GAN. Based on blinded evaluations of 7 cases, both radiologists ranked the Temporal-GAN higher than the 3D spatiotemporal GAN in 5 cases, and they were split in the remaining two cases (i.e. one ranked Temporal-GAN higher, and one ranked 3D spatiotemporal GAN higher in these two cases). SSIM (±SD), nRMSE (±SD)) which were calculated based on the testing dataset Group B1 for the Temporal-GAN, 3D spatiotemporal GAN, and the 2D GAN was (0.746±0.0495, 0.036±0.0072), (0.682±0.061, 0.053±0.010), and (0.481±0.0594, 0.072±0.0138), respectively. The mean of the normalized Tenengrad focus measure (±SD) for the reconstructed and respiratory motion-corrected results obtained by the Temporal-GAN and the 3D spatiotemporal GAN was 0.702±0.1408 and 0.762±0.146, respectively.

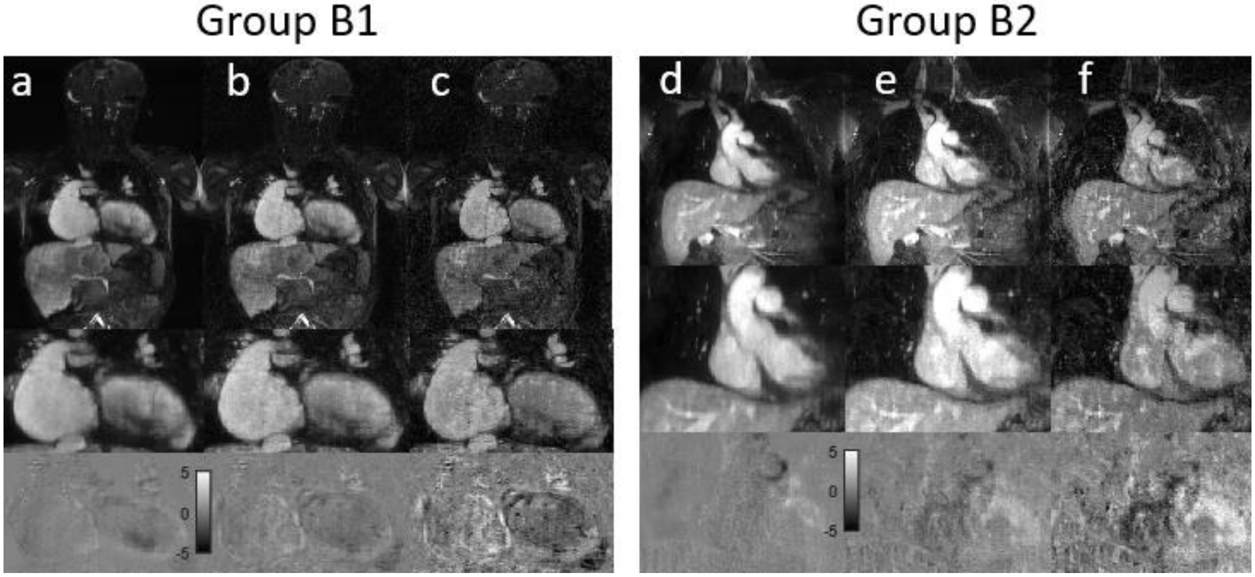


Figure SE1. Qualitative results obtained by three techniques for two patient cases selected from the testing datasets Group B1 and Group B2. It shows the reconstruction and respiratory motion correction results for the Temporal-GAN (a, d), 3D spatiotemporal GAN (b, e), and 2D GAN (c, f). The magnified heart region is shown for each image (2nd row of each panel). The bottom row of each panel shows the temporal difference maps between two sequential cardiac frames. Both Temporal-GAN and 3D spatiotemporal GAN achieved better results regarding aliasing and respiratory motion and flickering artifacts reduction than the 2D GAN.

F. Supporting Information Document S6 (Cardiorespiratory gated inputs vs. the cardiac gated inputs)

F1. Purpose: In this supplemental study, we sought to investigate the difference between TAV-GAN trained based on 1) cardiac-gated zero-filled images as input and cardiorespiratory-gated CS reconstruction as a reference, and the TAV-GAN trained based on 2) cardiorespiratory-gated zero-filled images as input and cardiorespiratory-gated CS reconstruction as a reference.

F2. Method: As illustrated in the main manuscript, TAV-GAN was trained based on the cardiac-gated zero-filled images as input and cardiorespiratory-gated CS reconstruction images as the target. Another TAV-GAN with the same training procedure and parameters was trained based on the cardiorespiratory-gated zero-filled images as input and cardiorespiratory-gated CS reconstruction images as the target. The performance of two TAV-GANs was compared qualitatively with regard to regular respiratory motion artifact and irregular respiratory motion artifact.

F3. Results: Figure SF1 shows the qualitative results obtained by SG CS WV (a, d), TAV-GAN trained based on the cardiac-gated zero-filled images as input (b, e), and TAV-GAN trained based on cardiorespiratory-gated zero-filled images as input (c, f) for two representative cases selected from Group B1 and Group B2. For the patient with regular breathing, there was no apparent difference between the two TAV-GANs – both of them provided good image qualities. For the Group B2 patient, the TAV-GAN trained based on the cardiac-gated zero-filled images as input provided better overall image quality.

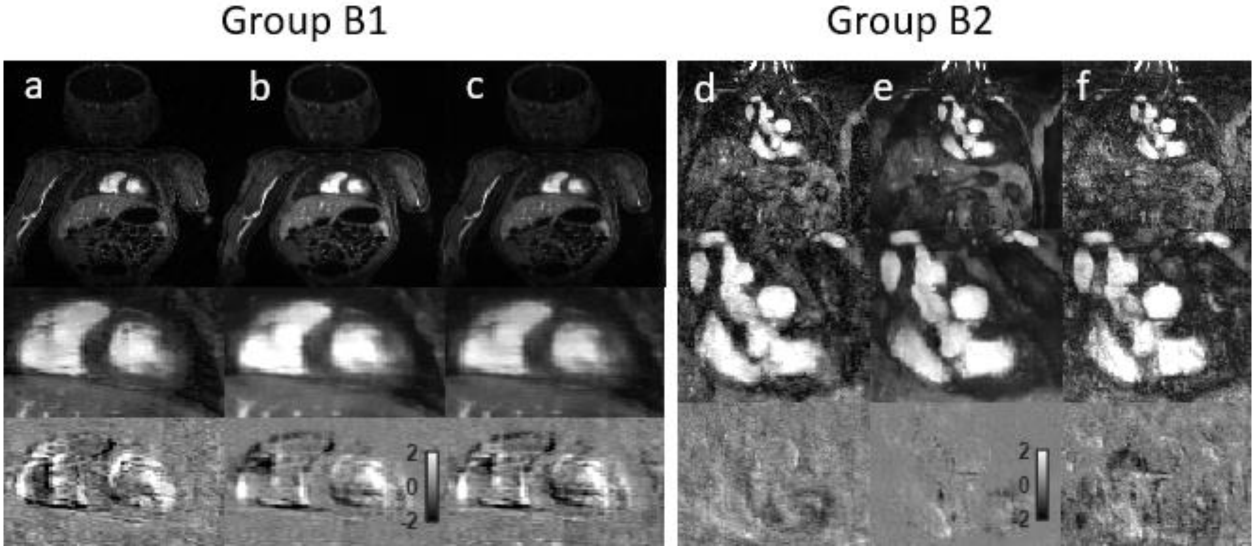


Figure SF1. Qualitative representative results of two unseen cases from Group B1 and Group B2. (a-c) show the un-aliased and respiratory artifact-corrected images from a patient with a regular respiratory pattern during scanning, obtained by SG CS-WV, TAV-GAN (trained based on cardiac-gated zero-filled images as the input), and TAV-GAN (trained based on cardiorespiratory gated zero-filled images as the input), respectively. (d-f) show images using the same techniques from a patient with irregular respiratory motion. The TAV-GAN trained based on the cardiorespiratory gated zero-filled images as the input would reduce the respiratory and aliasing artifacts in the case with regular breathing, but it seems in the case with irregular breathing, its performance dropped substantially. In each panel, the 2nd rows are amplified images of the heart region, and the third rows are temporal difference maps for two sequential cardiac phases.

F4. Discussion: For the presented test results (See Fig. SF1) from Group B1, which had regular breathing and was similar to the data in training datasets (Group A), both TAV-GANs showed similar performance. However, the TAV-GAN trained based on the cardiorespiratory gated zero-filled images, could not provide satisfactory results in the presence of irregular breathing (See Fig. SF1; Group B2). The TAV-GAN trained based on the cardiac gated zero-filled images as the input shows more robustness in the testing stage on the data with irregular breathing. We speculate that when the TAV-GAN is trained on cardiorespiratory-gated zero-filled images as the input, it would only learn how to remove under-sampling aliasing artifacts, which is easier than removing the aliasing and respiratory artifacts simultaneously. This drawback may compromise the network's ability in removing any residual motion after respiratory self-gating in the testing stage.

References

1. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process. 2004 Apr;13(4):600-12. doi: 10.1109/tip.2003.819861. PMID: 15376593.

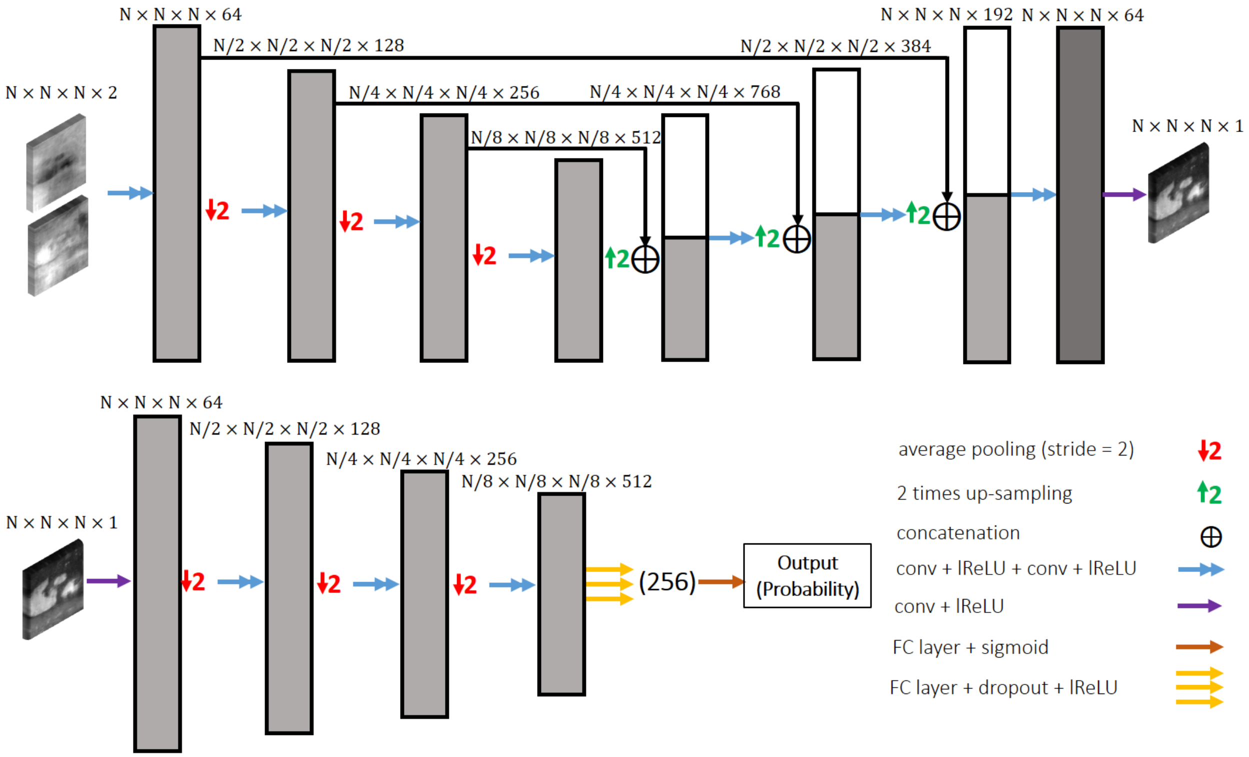
2. Han F, Zhou Z, Han E, et al. Self-gated 4D multiphase, steady-state imaging with contrast enhancement (MUSIC) using rotating cartesian K-space (ROCK): Validation in children with congenital heart disease. *Magn Reson Med*. 2017;78(2):472-483

3. Zhou Z, Han F, Yoshida T, Nguyen KL, Finn JP, Hu P. Improved 4D cardiac functional assessment for pediatric patients using motion-weighted image reconstruction. MAGMA. 2018 Dec;31(6):747-756. doi: 10.1007/s10334-018-0694-8. Epub 2018 Jul 24. PMID: 30043124.

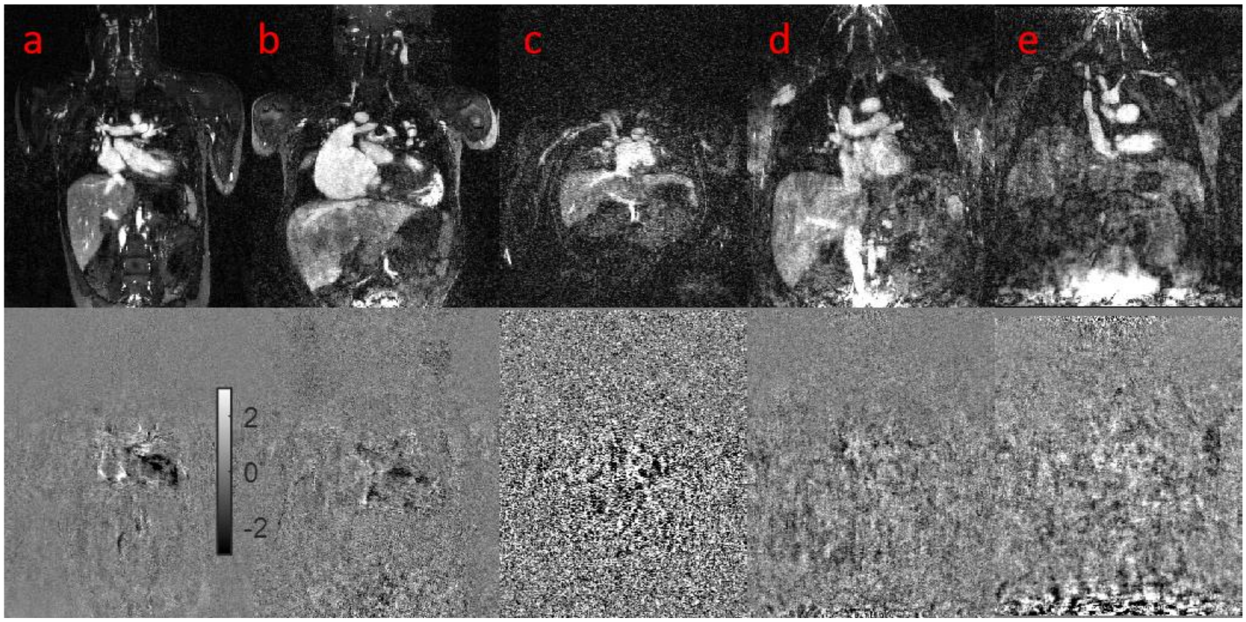
4. Krotkov E. Focusing. Int J Comput Vision 1, 223–237 (1988).

5. Hashim Mir, Peter Xu, Peter van Beek, "An extensive empirical evaluation of focus measures for digital photography," Proc. SPIE 9023, Digital Photography X, 90230I (7 March 2014).

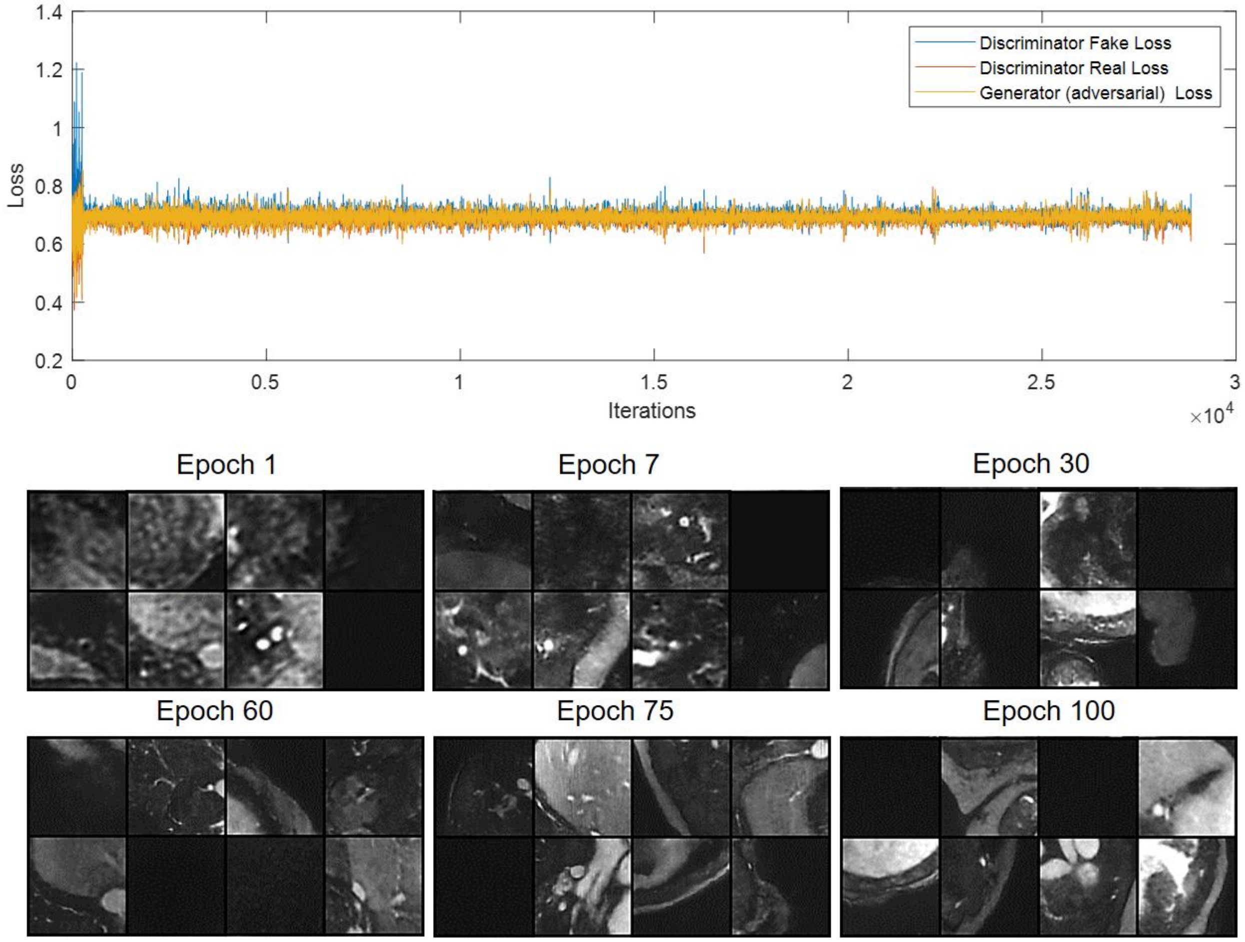
G. Supporting Figures



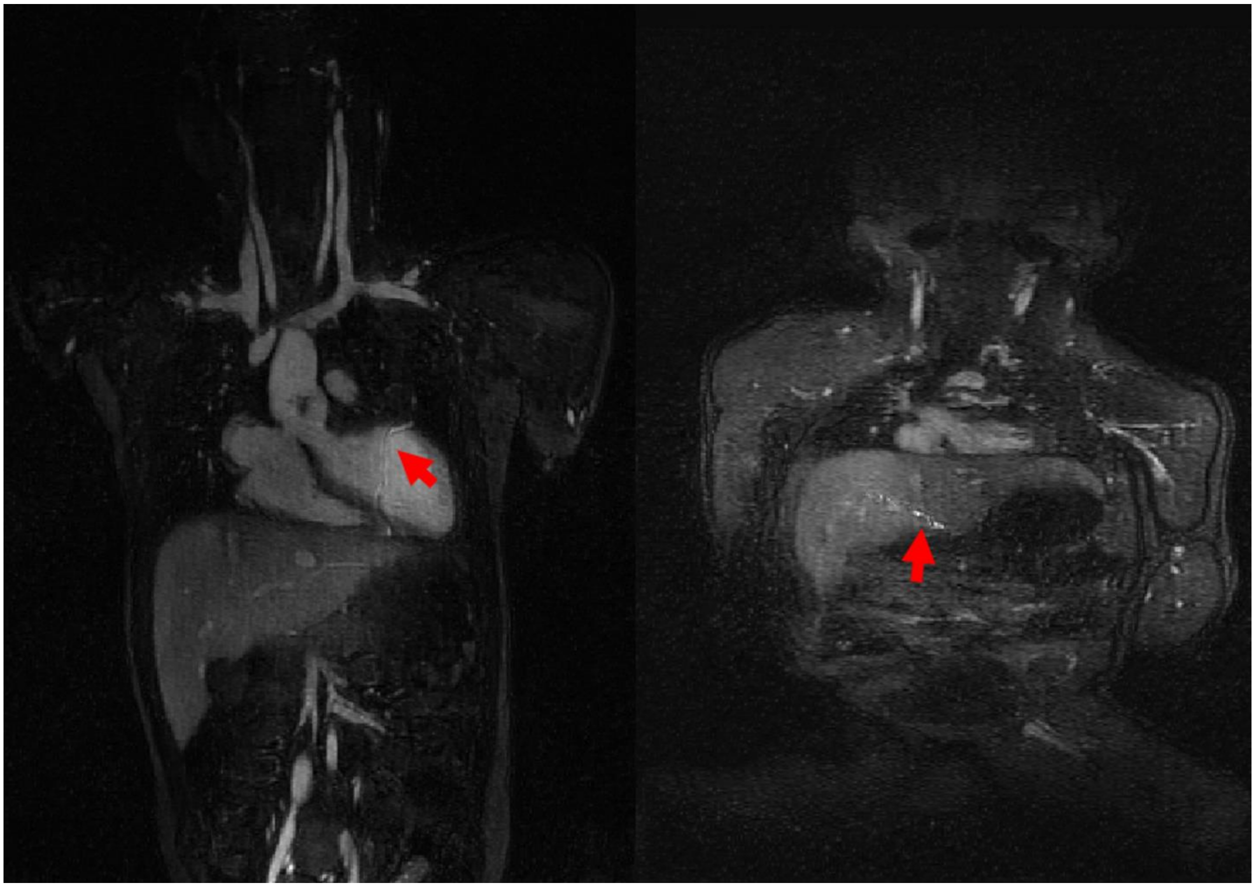
Supporting Information Figure S1. Detailed network structure for the volumetric generator and discriminator used in TAV-GAN. The generator is a 3D U-Net which consists of two paths: (I) the encoder path, which contains three downsampling blocks; (II) the decoder path, which includes three up-sampling blocks. Each block contains two convolutional layers, with each layer containing learnable convolution filters followed by Leaky ReLU (LReLU). Convolutional layers in the first block of the network contain 64 convolutional kernels, and the number of kernels doubles in each deeper block. Down-sampling and up-sampling blocks in the encoder and decoder paths are connected via average polling (strides = 2) and up-sampling (strides = 2). A skip connection is used to pass the data between each pair of same-sized up-sampling and down-sampling blocks. The discriminator is a binary classifier that contains three down-sampling operations followed by two convolutional layers in which each convolutional layer contains convolutional kernels followed by LReLU. The last two layers are the fully connected layer followed by dropout and LReLU, and a single decision fully connected layer with a sigmoid activation function. Discriminator takes the magnitude of the generated images to decide whether it is “generated” or “clean” images. The input and output of the generator for the Volumetric-GAN and temporally aware volumetric GAN (TAV-GAN) in the training phase are complexed-valued 3D image patches with size N×N×N×2 (real and imaginary), and magnitude-valued 3D image patches with size N×N×N×1, respectively. The input and output of the generator for the Temporal-GAN in the training phase are magnitude-valued 3D image patches with size N×N×N×3 (three sequential cardiac phases) and a magnitude-valued 3D image patch with size N×N×N×1, respectively. Due to the limitation of the GPU memory, N=64 is used in this work.



Supporting Information Figure S2. Representative examples for the datasets: columns (a), (b), (c-e) represent qualitative examples of the images from the dataset A (training dataset), dataset B1 (mild testing dataset), and dataset B2 (severe testing dataset), respectively. The first row shows the magnitude of a slice from the volumetric images, and the second row shows the difference map between two sequential cardiac phases. As can be seen in (a), it has the lowest noise and flickering artifacts through the cardiac phases among the others. The image in the column (b) has relatively higher noise and flickering artifacts through the cardiac phases than the image in column (a). Based on the calculation of the noise inside a 15×15×15 cubic region from the background, images in the datasets B1 (mean of the standard deviation = 0.076) are 2 times noisier than the images in the datasets A (mean of the standard deviation = 0.038). Column (c) presents image that was profoundly affected by noise. Approximately, the noise level for noisy images in datasets B2 (mean of the standard deviation = 0.304) based on the calculation of the noise inside a 15×15×15 cubic region from the background is, on average, 8 times the images in datasets A. Column (d) shows an image from a CHD patient with breathing irregularities scanned under anesthesia. As shown in column (d), image quality is degraded due to the respiratory motion artifacts. The image in column (e) shows an image from a CHD patient scanned under free-breathing without anesthesia. As shown in column (e), the quality of the image is degraded substantially due to the respiratory artifact and breathing irregularities.



Supporting Information Figure S3. Training convergence: first row plots the loss components versus the iterations for the generator and the discriminator of the temporally aware volumetric GAN (TAV-GAN). Only adversarial loss is plotted for the generator, and it means how well the generator can fool the discriminator. The discriminator contains two components associated with classification performance for both real and fake images. As seen in the first row, all three components converge to an equilibrium state (0.7). Besides, this convergence is happening very fast because of the practical training strategy introduced in this work. The second row shows the qualitative validation results through the epochs. It seems that after epoch 60 (15000 iterations), image quality is improved sufficiently.



Supporting Information Figure S4. Hallucination effect: by training the generative adversarial networks on the datasets with noisy ground truth, some characteristic artifacts were introduced to the image. As pointed with the red arrow, such a network generated spurious artifact has appeared in the left myocardium and liver region. For this case, we trained the network on dataset B1 and tested it on dataset A. We note that on average, the dataset B1 was two times noisier than the dataset A. This result reveals the importance of curating the data and using less noisy target reference images for training GANs. Otherwise, spurious features might be introduced to the reconstructed images.

H. Supporting Tables

Supporting Information Table S1. Multiple comparisons between the quantitative SSIM *score* of the images reconstructed by different methods. At the α=0.05 level of significance, SSIM *scores* of the images reconstructed by each of the 3D network (Volumetric-GAN, Temporal-GAN, 3D U-Net, TAV-GAN) are higher than the images reconstructed by 2D GAN. The paired SSIM score differences among the 3D networks were non-significant.

Comparison Method	**(I) Groups**	**(J) Groups**	**Mean Difference (I-J)**	**Sig.**	**95% Confidence Interval**	
					Lower Bound	**Upper Bound**
Tukey HSD	2D GAN	ZF	0.1054*	0.000	0.041	0.169
		Temporal-GAN	-0.2652*	0.000	-0.329	-0.201
		TAV-GAN	-0.3037*	0.000	-0.368	-0.240
		Volumetric-GAN	-0.2705*	0.000	-0.334	-0.206
		3D U-Net	-0.2511*	0.000	-0.315	-0.187
	ZF	2D GAN	-0.1054*	0.000	-0.169	-0.041
		Temporal-GAN	-0.3706*	0.000	-0.435	-0.306
		TAV-GAN	-0.4091*	0.000	-0.473	-0.345
		Volumetric-GAN	-0.3759*	0.000	-0.440	-0.312
		3D U-Net	-0.3566*	0.000	-0.421	-0.292
	Temporal-GAN	2D GAN	0.2652*	0.000	0.201	0.329
		ZF	0.3706*	0.000	0.306	0.435
		TAV-GAN	-0.0385	0.490	-0.102	0.025
		Volumetric-GAN	-0.0053	1.000	-0.069	0.059
		3D U-Net	0.0140	0.987	-0.050	0.078
	TAV-GAN	2D GAN	0.3037*	0.000	0.240	0.368
		ZF	0.4091*	0.000	0.345	0.473
		Temporal-GAN	0.0385	0.490	-0.025	0.102
		Volumetric-GAN	0.0332	0.646	-0.031	0.097
		3D U-Net	0.0525	0.166	-0.011	0.117
	Volumetric-GAN	2D GAN	0.2705*	0.000	0.206	0.334
		ZF	0.3759*	0.000	0.312	0.440
		Temporal-GAN	0.0053	1.000	-0.059	0.069
		TAV-GAN	-0.0332	0.646	-0.097	0.031
		3D U-Net	0.0193	0.947	-0.045	0.083
	3D U-Net	2D GAN	0.2511*	0.000	0.187	0.315
		ZF	0.3566*	0.000	0.292	0.421
		Temporal-GAN	-0.0140	0.987	-0.078	0.050
		TAV-GAN	-0.0525	0.166	-0.117	0.011
		Volumetric-GAN	-0.0193	0.947	-0.083	0.045

*. The mean difference is significant at the 0.05 level. Tukey HSD = Tukey honestly significant difference

Supporting Information Table S2. Multiple comparisons between the quantitative nRMSE *score* of the images reconstructed by different methods. At the α=0.05 level of significance, nRMSE *scores* of the images reconstructed by each of the 3D network (Volumetric-GAN, Temporal-GAN, 3D U-Net, TAV-

GAN) are lower than the images reconstructed by 2D GAN. The paired nRMSE differences among the 3D networks were non-significant.

Comparison Method			Mean		95% Confidence Interval	
	(I) Groups	(J) Groups	Difference (I-J)	Sig.	Lower Bound	Upper Bound
Tukey HSD	2D GAN	ZF	-0.0218*	0.001	-0.037	-0.006
		Temporal-GAN	0.0358*	0.000	0.020	0.051
		TAV-GAN	0.0423*	0.000	0.027	0.058
		Volumetric-GAN	0.0339*	0.000	0.018	0.049
		3D U-Net	0.0321*	0.000	0.017	0.047
	ZF	2D GAN	0.0218*	0.001	0.006	0.037
		Temporal-GAN	0.0577*	0.000	0.042	0.073
		TAV-GAN	0.0641*	0.000	0.049	0.079
		Volumetric-GAN	0.0557*	0.000	0.040	0.071
		3D U-Net	0.0539*	0.000	0.038	0.069
	Temporal-GAN	2D GAN	-0.0358*	0.000	-0.051	-0.020
		ZF	-0.0577*	0.000	-0.073	-0.042
		TAV-GAN	0.0064	0.815	-0.009	0.022
		Volumetric-GAN	-0.0019	0.999	-0.017	0.013
		3D U-Net	-0.0038	0.978	-0.019	0.011
	TAV-GAN	2D GAN	-0.0423*	0.000	-0.058	-0.027
		ZF	-0.0641*	0.000	-0.079	-0.049
		Temporal-GAN	-0.0064	0.815	-0.022	0.009
		Volumetric-GAN	-0.0084	0.595	-0.024	0.007
		3D U-Net	-0.0102	0.374	-0.025	0.005
	Volumetric-GAN	2D GAN	-0.0339*	0.000	-0.049	-0.018
		ZF	-0.0557*	0.000	-0.071	-0.040
		Temporal-GAN	0.0019	0.999	-0.013	0.017
		TAV-GAN	0.0084	0.595	-0.007	0.024
		3D U-Net	-0.0018	0.999	-0.017	0.013
	3D U-Net	2D GAN	-0.0321*	0.000	-0.047	-0.017
		ZF	-0.0539*	0.000	-0.069	-0.038
		Temporal-GAN	0.0038	0.978	-0.011	0.019
		TAV-GAN	0.0102	0.374	-0.005	0.025
		Volumetric-GAN	0.0018	0.999	-0.013	0.017

*. The mean difference is significant at the 0.05 level. Tukey HSD = Tukey honestly significant difference

Supporting Information Table S3. Multiple comparisons between the quantitative normalized Tenengrad focus measure of the images reconstructed by different methods. At the $\alpha=0.05$ level of significance, sharpness *scores* of the images reconstructed by Volumetric-GAN, TAV-GAN, and Temporal-GAN, are higher than the images reconstructed by 3D U-Net. 2D GAN is excluded from the sharpness analysis because of the sensitivity of the Tenengrad focus measure to the residual high-frequency artifacts.

		Mean		95% Confidence Interval	
(I) group	(J) group	Difference (I-J)	Sig.	Lower Bound	Upper Bound
TAV-GAN	Volumetric-GAN	-0.0064	0.999	-0.142	0.129
	Temporal-GAN	0.1196	0.101	-0.016	0.255
	3D U-Net	0.5353*	0.000	0.399	0.671
Volumetric-GAN	TAV-GAN	0.0064	0.999	-0.129	0.142
	Temporal-GAN	0.1261	0.077	-0.010	0.262
	3D U-Net	0.5418*	0.000	0.406	0.677
Temporal-GAN	TAV-GAN	-0.1196	0.101	-0.255	0.016
	Volumetric-GAN	-0.1261	0.077	-0.262	0.010
	3D U-Net	0.4157*	0.000	0.280	0.551
3D U-Net	TAV-GAN	-0.5353*	0.000	-0.671	-0.399
	Volumetric-GAN	-0.5418*	0.000	-0.677	-0.406
	Temporal-GAN	-0.4157*	0.000	-0.551	-0.280

*. The mean difference is significant at the 0.05 level.

Supporting Information Table S4. Multiple comparisons of subjective image quality rank comparisons were performed in Stage 1 subjective image quality evaluation. Among the 6 techniques ranked, only four techniques (Volumetric-GAN, Temporal-GAN, 3D U-Net, and self-gated CS-WV) are shown. We excluded TAV-GAN from this analysis because of its outstanding scores in the rank comparison, and it was consistently ranked highest among the 6 techniques. We also excluded the 2D GAN in this analysis because it was ranked consistently the worst among the 6 techniques. We excluded 2D GAN to ensure that the assumption of the variance's homogeneity is valid for the Tukey HSD test. At the α=0.05 level of significance, images reconstructed by 3D U-Net had lower scores in comparison to the Temporal-GAN. Mean difference values indicated that the Temporal-GAN has a higher rank score than other methods, including Volumetric-GAN, SG CS-WV, and 3D U-Net, although the difference was not significant.

Comparison Method	(I) Method	(J) Method	Mean Difference (I-J)	Sig.	95% Confidence Interval	
					Lower Bound	Upper Bound
Tukey HSD	Temporal-GAN	Volumetric-GAN	0.250	0.155	-0.06	0.56
		SG CS-WV	0.233	0.204	-0.07	0.54
		3D U-Net	0.417*	0.003	0.11	0.72
	Volumetric-GAN	Temporal-GAN	-0.250	0.155	-0.56	0.06
		SG CS-WV	-0.017	0.999	-0.32	0.29
		3D U-Net	0.167	0.496	-0.14	0.47
	SG CS-WV	Temporal-GAN	-0.233	0.204	-0.54	0.07
		Volumetric-GAN	0.167	0.999	-0.29	0.32
		3D U-Net	0.183	0.411	-0.12	0.49
	3D U-Net	Temporal-GAN	-0.417*	0.003	-0.72	-0.11
		Volumetric-GAN	-0.167	0.496	-0.47	0.14
		SG CS-WV	-0.183	0.411	-0.49	0.12

*. The mean difference is significant at the 0.05 level. Tukey HSD = Tukey honestly significant difference