A. Training and Validation

A.1 Training Details

For training, we prepare paired (slice stack, motion stack) and (splatted volume, ground truth volume) training data by applying a random isotropic zoom (±26 voxels), horizontal flip, rotation (Euler angles ±20º for adult brains and ±180º for fetal) and translation (±13 voxels) to the registered 256^3 reference volume (1mm^3 acquisition, Talairach for adult and 0.8mm^3 FeTA reconstruction, CRL for fetal) to first simulate the diversity in scanned subjects and various poses that they may be scanned under. To simulate slicing, we apply random rotations (Euler angles ±20º) and translations (±26 voxels) that vary smoothly across slices by randomly sampling 32 to 64 rotations and translations from the ranges shown above and smoothly interpolating them using cubic B splines. We interleave the two halves of motion trajectories to simulate two-shot sequences typically used in 2D MRI acquisitions.

Slice acquisition is simulated by blurring the slices using a boxcar PSF (four voxels wide) along the slicing direction (sagittal, axial or coronal for adult brains, and sagittal in the case of fetal, where the ±180º Euler angles mean that slices are acquired along random directions anyway) and sampling every fourth slice along the same axis. Slice intensities are manipulated by applying Gaussian noise with noise standard deviation $\sigma = 0.01$ and gamma augmentation with exponent $\gamma \in [0.9,1]$. Finally, the acquired slices are replicated along the slicing direction by a factor of four again so that the slice stacks have isotropic in- and through-plane resolutions. We subsample the slice stacks by a factor of two (i.e., to 2mm^3 or 1.6mm^3) to train the slice motion prediction network. No subsampling is performed when training the interpolator.

We train our slice motion and interpolation networks for $256{,}000$ steps using ADAM, with an initial learning rate of 10^{-4}, which is reduced to 0 with poly scheduling (exponent of 0.9), weight decay of 0 and momentum of 0.90. We mask out background voxels when computing the training loss. In the case of fetal SVR, we over-sample the FeTA portion of the training data fivefold, and the CRL portion tenfold. We simply pick the model of the last epoch for model selection but still monitor validation metrics for potential overfitting.

A.2 Validation Metrics

Similar to our training loss (9), all our validation metrics compensate for any global rigid motion offset may that exist between predicted and true slice motion; see (9). In addition to the MSE $\mathcal{L}_{\text{MSE}}(\mathbf{u},\mathbf{y}) = (1/N)\|\mathbf{u}-\mathbf{y}\|_F^2$ of the predicted slice motion $\mathbf{u} \in \mathbb{R}^{N\times 3}$ w.r.t. true motion $\mathbf{y} \in \mathbb{R}^{N\times 3}$, we use the average end-point error (EPE) metric [33], defined as

$$\mathcal{L}_{\text{EPE}}(\mathbf{u},\mathbf{y}) = (1/N)\|\mathbf{u}-\mathbf{y}\|_{2,1}, \tag{A1}$$

that is, the mean Euclidean distance between the end-points of two slice motion fields (both metrics shown here without rigid compensation for clarity). In previous work [20, 25], a similar metric is proposed to measure the average Euclidean distance between the predicted and true slice positions at the anchor points of the slices (anchor point error, APE):

$$\mathcal{L}_{\text{APE}}(\mathbf{u},\mathbf{y}) = (1/3)\|\mathbf{u}_{\{0,1,2\}}-\mathbf{y}_{\{0,1,2\}}\|_{2,1}, \tag{A2}$$

in which $\mathbf{u}_{\{0,1,2\}}$ and $\mathbf{y}_{\{0,1,2\}}$ denote the position vectors at the anchor points of the grids that define the voxel locations of the respective slices in 3D space. Typically, anchor points are assumed to be at the center, bottom left and bottom right corners of a given slice. Assuming that slices are undergoing rigid motion, the APE is equivalent to the EPE averaged on the right-triangular region formed by the three anchor points on the reference slice.

A.3 SVRnet Model Implementation

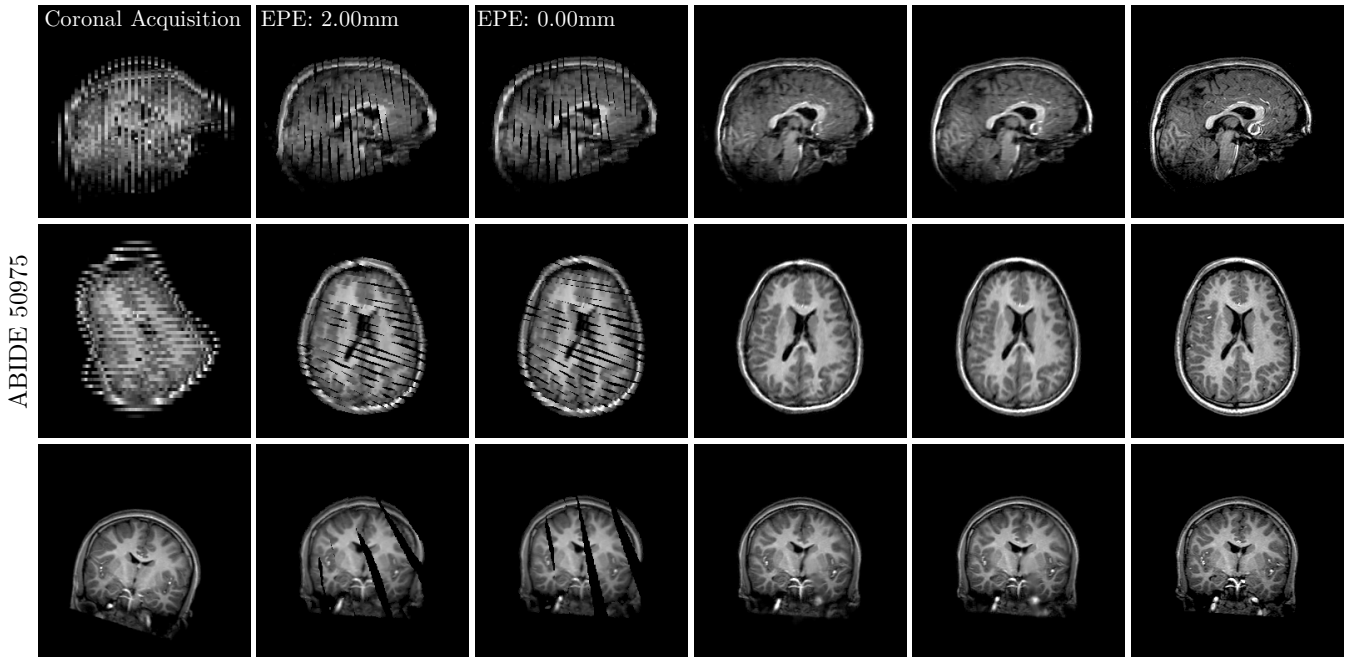
For reproducibility, we port the original TensorFlow 1.13 implementation of SVRnet [20] with an Inception backbone to PyTorch 1.13, where we use a ResNet-34 backbone and a prediction head consisting of a 512×9 dense layer, which predicts the slice position vectors at three anchor points. We find that 2D batch normalization based on collected statistics does not perform well at test time and opt to normalize each example based on the statistics of each slice stack. We use subsampled slices of size 128×128 pixels and interpolate the predicted anchor point position vectors to a linear motion field and subtract the slice voxel coordinates to output slice motion. We train this implementation of SVRnet using the regular MSE loss on output slice motion field. We initialize the model with the torchvision ImageNet1K_v1 weights.

A.4 SVoRT Model Configuration

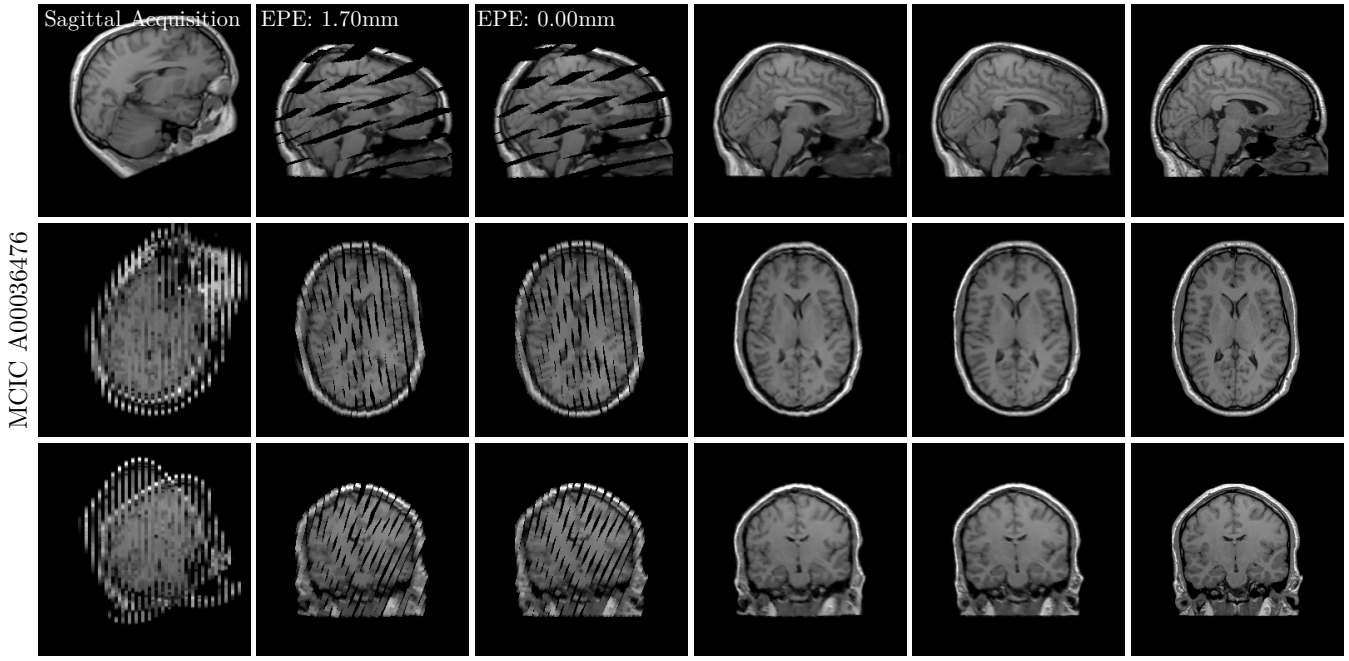
For comparison with SVoRT (v2), we use model weights provided by Xu et al. [25] on their repository, configure the model to use one slice stack with a slice gap of 3.2mm, and optimize the reconstruction PSF (slice thickness of 1.6mm) for validation accuracy (i.e., average motion end-point error) on the 12 FeTA validation subjects using an exponential grid search. We fixed the stack positional encoding of SVoRT to 0 (rather than a random integer) for reproducible results. We convert SVoRT's transform output to dense motion fields to compute the motion MSE and EPE for validation accuracy.

B. Additional SVR Results

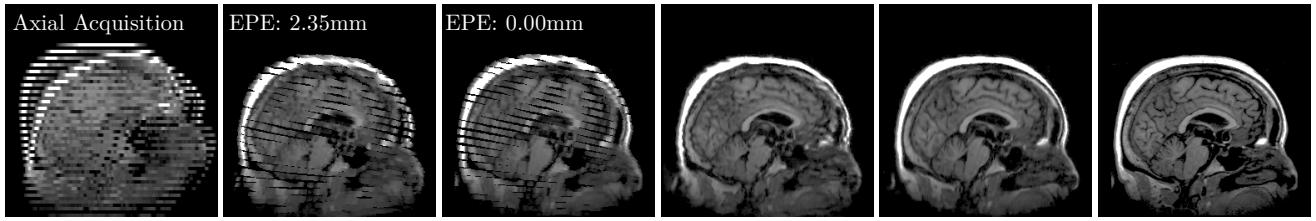
Here, we provide additional SVR results on our adult and fetal datasets. Figure B1 visualizes reconstructions of adult brain MR volumes for three of our validation subjects, with all three orthogonal views shown for completeness. Figure B2 similarly visualizes fetal reconstructions. We include the corresponding SVoRTv2 results for comparison, noting that SVRnet reconstructions are garbled in many cases (see first row of Figure 9) and are less meaningful to compare against.

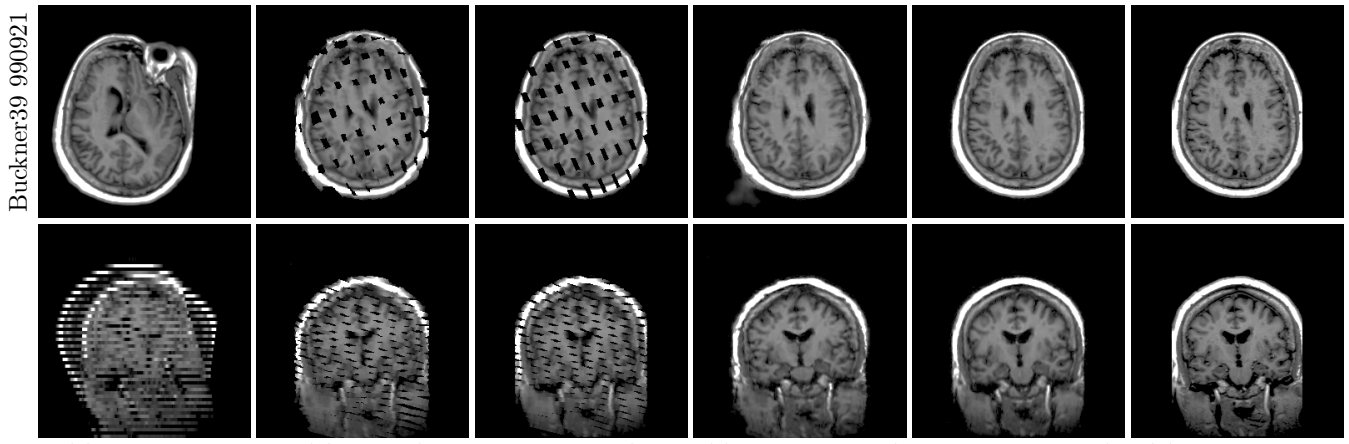


(a) Slice Stack (b) Splat (Ours, True Motion) (c) Interpolated (Ours, True Motion) (d) True Volume



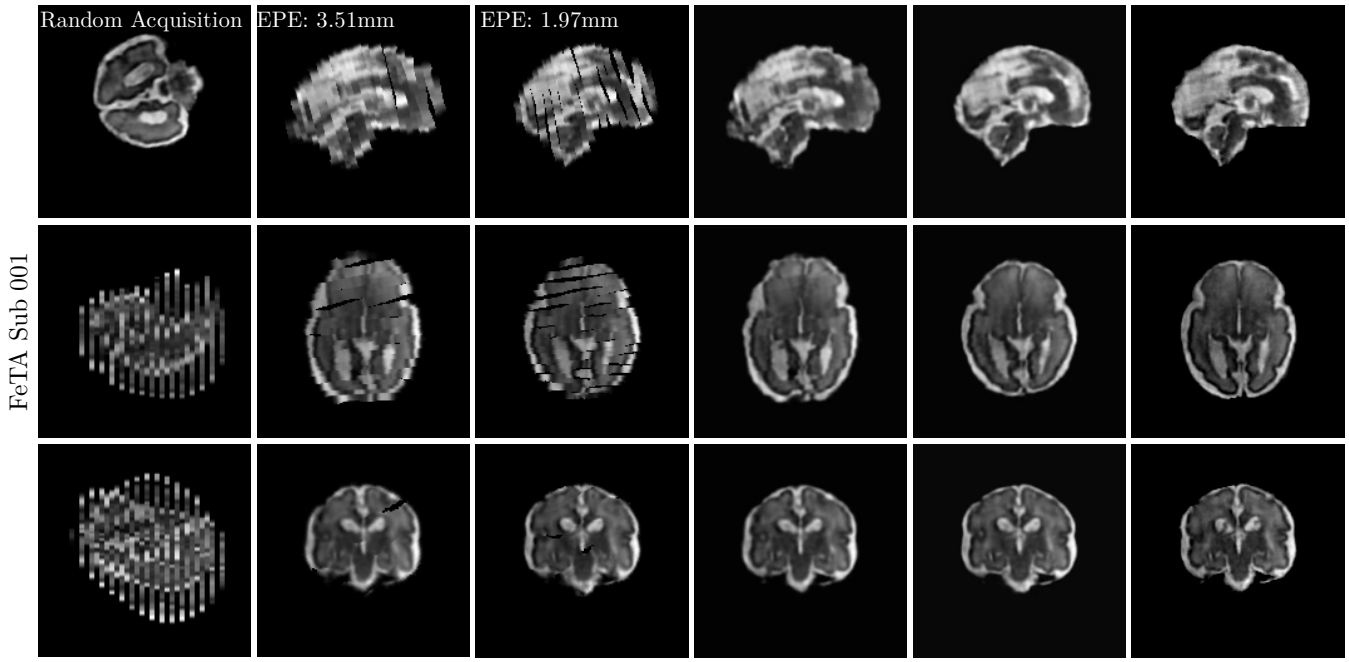
(a) Slice Stack (b) Splat (Ours, True Motion) (c) Interpolated (Ours, True Motion) (d) True Volume



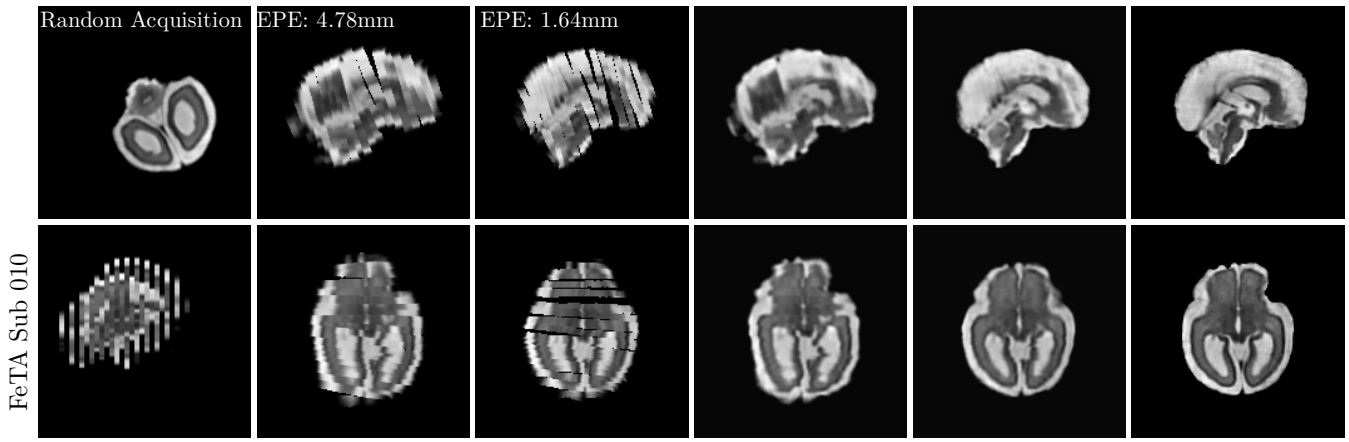


(a) Slice Stack (b) Splat (Ours, True Motion) (c) Interpolated (Ours, True Motion) (d) True Volume

Figure B1: SVR of adult brain scans. We visualize our SVR results on slice stacks synthesized using random slice motion (a). Using the predicted motion stack, we splat slice data to reconstruct the underlying 3D volume (b). We interpolate the missing intensities (holes) in our reconstruction (c). We additionally visualize in (b) and (c) splat and interpolated results obtained when the true motion stack is used.



(a) Slice Stack (b) Splat (SVoRTv2, Ours) (c) Interp (SVoRTv2, Ours) (d) True Volume



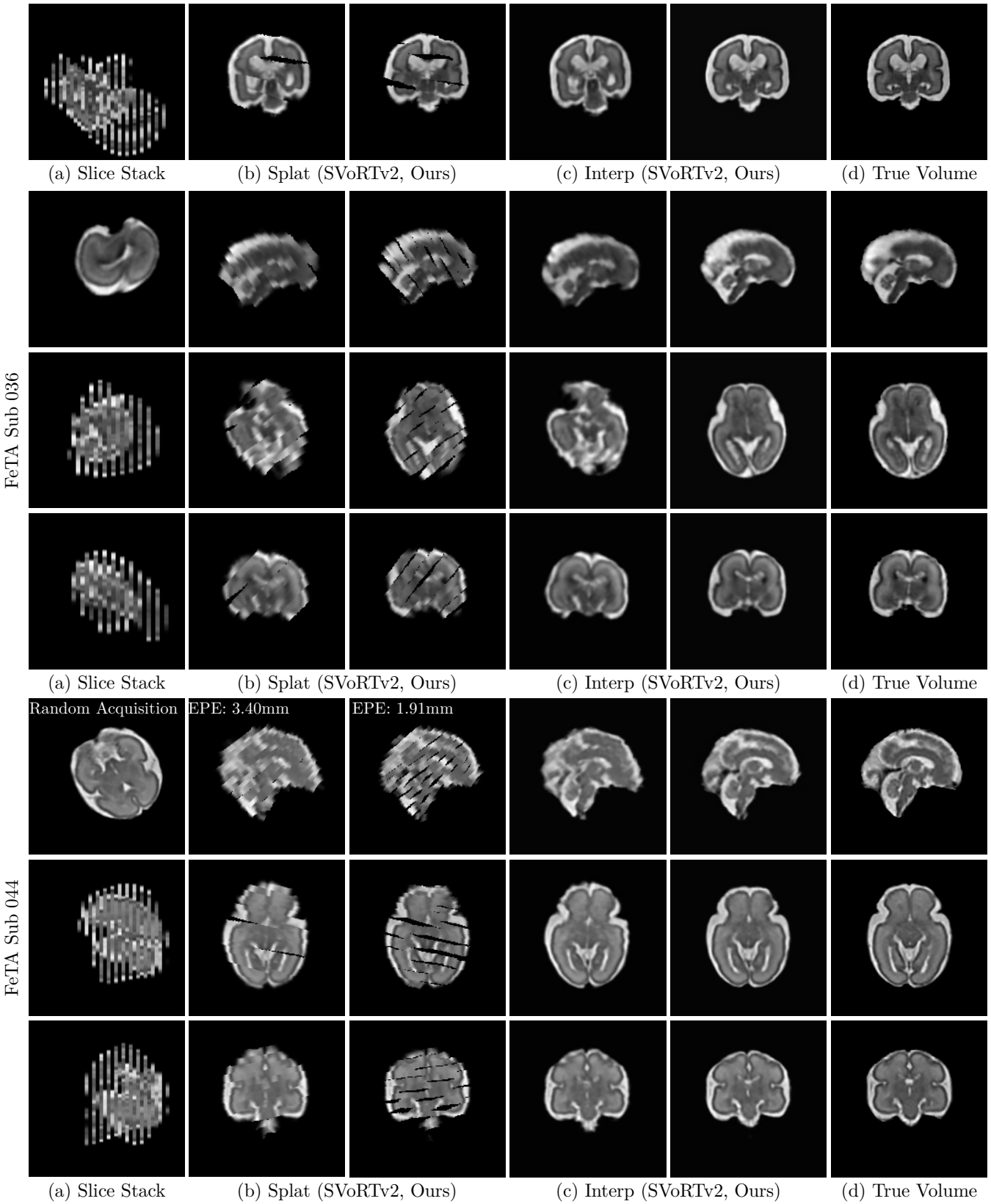


Figure B2: Single-stack fetal SVR. We visualize the SVR results on validation subjects from the FeTA dataset [73]. Our results closely resemble the ground truth volumes while SVoRTv2 reconstructions (with our interpolation applied) exhibit spatial distortion from inaccurate slice alignment.