Fast Cross-Strength Multi-Contrast Brain MRI Translation using Latent Bridge Matching

Siddharth Srivastava and Till Bretschneider

University of Warwick, United Kingdom
{Siddharth.Srivastava,T.Bretschneider}@warwick.ac.uk

Abstract. Magnetic Resonance Imaging (MRI) acquired at different field strengths exhibits pronounced variation in noise, resolution, homogeneity, and contrast, which limits comparability across acquisition settings and complicates downstream analysis. We address this with a unified conditional model for controllable field-to-field synthesis, built on the framework of conditional latent bridge matching. Our single model achieves highly competitive results across the validation phase for all three tasks of the MRIxFields2026 challenge without task-specific architectures or training. We achieve fast generation with only a single inference step, producing all modality and field-strength combinations for 30 axial slices in under 90 seconds, as well as cross-modality-strength translation for a full volume in under 70 seconds, on a single NVIDIA A5000 GPU. We further provide extensive ablations regarding different components of our solution.

Keywords: MRI harmonisation, bridge matching

1 Introduction

Magnetic Resonance Imaging (MRI) is a widely used method for non-invasive medical 3D imaging, providing both tissue and anatomical contrast for clinical diagnosis and medical research. MRI quality is highly dependent on the magnetic field strength of the acquisition scanner, with strengths ranging from ultra-low-field portable scanners at 0.1T, to clinical scanners in point-of-care settings at 1.5–3T, to ultra-high-field research scanners at 7T. Images acquired at different strengths vary substantially regarding geometric artefacts, signal-to-noise, resolution, homogeneity, and contrast, significantly affecting multi-site studies and the clinical translation of downstream machine learning models trained on such varied data.

Generative machine learning models based on flow or diffusion frameworks have the potential to infer high-field-equivalent anatomical structures from low-field inputs, overcoming physical hardware gaps. The Generalizable Cross-Field MRI Translation and Harmonization Challenge (MRIxFields2026) aims to provide a single benchmark for cross-field harmonisation and synthesis and advance

Code: https://gitlab.com/siddharthsrivastava/mrixfields-2026

multimodal clinical translation. The dataset consists of MRI volumes from 0.1T to 7T using T1, T2, and T2FLAIR modalities, split into two partitions: a large unpaired set of 1,900+ volumes, and a smaller paired set of 40 travelling-cohort volunteers (3 for training, 17 for validation, and 20 for testing); each volunteer in the paired set is scanned across all modalities and field strengths.

We propose a single unified conditional framework for all three tasks of the MRIxFields2026 challenge: ultra-high-field (7T) synthesis from arbitrary input field strengths (0.1T, 1.5T, 3T, 5T), higher-field (1.5T, 3T, 5T, 7T) generation from ultra-low-field MRI, and controllable field-to-field MRI synthesis. Our method encodes the source, target, and auxiliary modality slices into a shared latent space using a variational encoder; a separate lightweight condition encoder encodes the source field strength, target field strength, and target modality. A drift model then learns the conditional latent flow to transport the source latent toward the target latent; the resulting target latent is decoded back to pixel space using a variational decoder. At inference time, our method predicts the target latent in a single ODE step, enabling fast image synthesis: in practice, we generate all 4×3 field and modality combinations for 30 axial slices of a single volume in approximately 90 seconds on a single NVIDIA A5000 GPU.

2 Background

2.1 Rectified Flow and Bridge Matching

Rectified flow [7] and bridge matching [2] are methods that transport a source distribution onto a target distribution by constructing deterministic (stochastic in the bridge matching case) interpolants between samples and estimating the drift of the ODE (SDE in the bridge matching case) of the process using a neural network. Consider two distributions π_0 and π_1 from which we sample $X_0 \sim \pi_0$, $X_1 \sim \pi_1$ independently. Rectified flow constructs the linear interpolant

$$X_t = (1-t)X_0 + tX_1, \qquad t \in [0,1]$$

which recovers X_0 at $t = 0$ and X_1 at $t = 1$. The time derivative of X is constant along the interpolant

$$\dot{X}_t = \frac{\mathrm{d}}{\mathrm{d}t}[X_t] = X_1 - X_0, \tag{1}$$

and describes the velocity to follow a straight path [7]. This velocity field is modelled as a neural network $v_\theta(X,t)$ using a least-squares objective:

$$\theta^\star := \arg\min_\theta \mathbb{E}_{t,X_0,X_1}\left[\|v_\theta(X,t) - (X_1 - X_0)\|^2\right], \qquad t \sim \tau(t), \tag{2}$$

where $\tau(t)$ is the timestep distribution – whose minimiser is the conditional mean of the line directions $X_1 - X_0$ that pass through x at time t

$$v_{\theta^\star}(x,t) \approx \mathbb{E}[X_1 - X_0 \mid X_t = x].$$

Bridge matching [2] instead models the stochastic interpolant X_t as a general Brownian bridge between endpoints X_0 and X_1:

$$X_t = (1-t)X_0 + tX_1 + \sigma B_t, \qquad t \in [0,1] \tag{3}$$

where B_t is a standard Brownian bridge process and $\sigma \geq 0$ is the diffusion coefficient; note that $B_t = \sqrt{t(1-t)}\epsilon, \epsilon \sim \mathcal{N}(0, I)$ using the reparameterisation trick, and is pinned at both ends $B_0 = B_1 = 0$. In this case, the dynamics are governed by a stochastic differential equation (SDE), rather than an ordinary differential equation (ODE) as in eq. (1):

$$\mathrm{d}X_t = \frac{(X_1 - X_t)}{1-t}\,\mathrm{d}t + \sigma \mathrm{d}W_t \tag{4}$$

where W_t is a Brownian motion. As before, we fit a neural network $v_\theta(X, t)$ (the *drift* network) to model the drift term using least squares,

$$\arg\min_\theta \mathbb{E}_{t,X_0,X_1}\left[\left\| v_\theta(X_t,t) - \frac{(X_1 - X_t)}{1-t}\right\|^2\right] \tag{5}$$

which recovers the marginal drift $\mathbb{E}\left[\frac{X_1 - X_t}{1-t}\middle| X_t\right]$ independent of X_1.

2.2 Inference

After training v_θ using eq. (5), it approximates the marginal $\mathbb{E}\left[\frac{X_1 - X_t}{1-t}\middle| X_t\right]$, which no longer depends on the unknown endpoint X_1 as in eq. (4). Therefore, we can generate a sample from π_1 by first sampling $X_0 \sim \pi_0$ and integrating the learnt SDE

$$\mathrm{d}X_t = v_\theta(X_t, t)\,\mathrm{d}t + \sigma\,\mathrm{d}W_t$$

from $t = 0$ to $t = 1$, where W_t is a standard Brownian motion. This integral can be discretised and solved using Euler–Maruyama: on an interval $0 = \tau_0 < \tau_1 < \cdots < \tau_N = 1$ with step size $\Delta t = \tau_{n+1} - \tau_n$, we initialise $\hat{X}_0 \sim \pi_0$ and iterate

$$\hat{X}_{n+1} = \hat{X}_n + v_\theta(\hat{X}_n, \tau_n)\,\Delta t + \sigma\,\Delta W_n, \qquad \Delta W_n \sim \mathcal{N}(0, \Delta t), \tag{6}$$

where the increments ΔW_n are those of a Brownian motion rather than a bridge, since the endpoint is no longer fixed at inference. The final iterate $\hat{X}_1$ approximates a sample drawn from π_1.

3 Method

Performing bridge matching directly in pixel space is computationally expensive at high resolution. We therefore adapt the latent-diffusion paradigm [10] and perform bridge matching in a smaller learnt latent space, called *latent* bridge matching [2]. Specifically, an encoder maps an image to a lower-dimensional

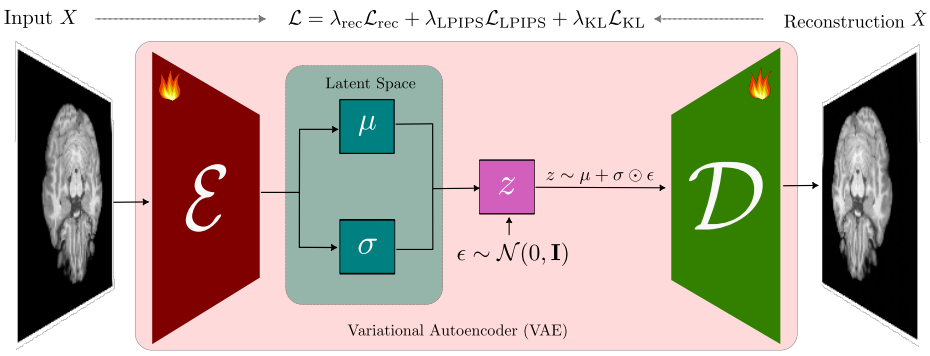


Fig. 1: Training pipeline for the variational autoencoder.

latent space, the generative process is run in latent space, and a decoder reconstructs the image in pixel space.

In our approach we finetune a pretrained variational autoencoder (VAE) [4], consisting of an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$, on 2D axial slices from the large unpaired dataset. Formally, the samples $X_0 \sim \pi_0$ and $X_1 \sim \pi_1$ are encoded to a latent space $Z_0 := \mathcal{E}(X_0), Z_1 := \mathcal{E}(X_1)$. Then, we adapt eq. (3) to form the *latent* stochastic interpolant Z_t

$$Z_t = (1-t)Z_0 + t(Z_1) + \sigma B_t, \qquad t \in [0,1] \tag{7}$$

which then leads to the latent bridge matching (LBM) loss based on eq. (5)

$$\mathcal{L}_{\text{LBM}} = \mathbb{E}_{t\sim\tau(t),X_0,X_1}\left[\left\| v_\theta(Z_t,t) - \frac{(Z_1 - Z_t)}{1-t}\right\|^2\right]. \tag{8}$$

Inference in latent space follows section 2.2, except that we use the dynamics of $\mathrm{d}Z_t = v_\theta(Z_t,t)\,\mathrm{d}t + \sigma\,\mathrm{d}W_t$, and the final approximation $\hat{Z}_1$ is decoded out of latent space as $\hat{X}_1 := \mathcal{D}(\hat{Z}_1)$.

3.1 Models

Variational Autoencoder. We initialise a variational autoencoder from the `sd-research/stable-diffusion-2-1-base` checkpoint and finetune on the unpaired dataset: we showcase this training in Figure 1. Given an input slice X, the encoder predicts the parameters of the approximate Gaussian posterior parameterised by μ and σ, from which the latent code z is sampled using the reparameterisation trick [4] and passed to the decoder for reconstructing $\hat{X}$. The VAE is optimised end-to-end using the objective

$$\mathcal{L} = \lambda_{\text{rec}}\mathcal{L}_{\text{rec}} + \lambda_{\text{LPIPS}}\mathcal{L}_{\text{LPIPS}} + \lambda_{\text{KL}}\mathcal{L}_{\text{KL}} \tag{9}$$

where $\mathcal{L}_{\text{rec}}$ is the pixel-wise reconstruction loss, $\mathcal{L}_{\text{LPIPS}}$ is the learned perceptual image patch similarity (LPIPS) [14] loss, and $\mathcal{L}_{\text{KL}}$ is the reverse KL divergence [6] between $\mathcal{N}(\mu,\sigma^2)$ and $\mathcal{N}(0,\mathbf{I})$.

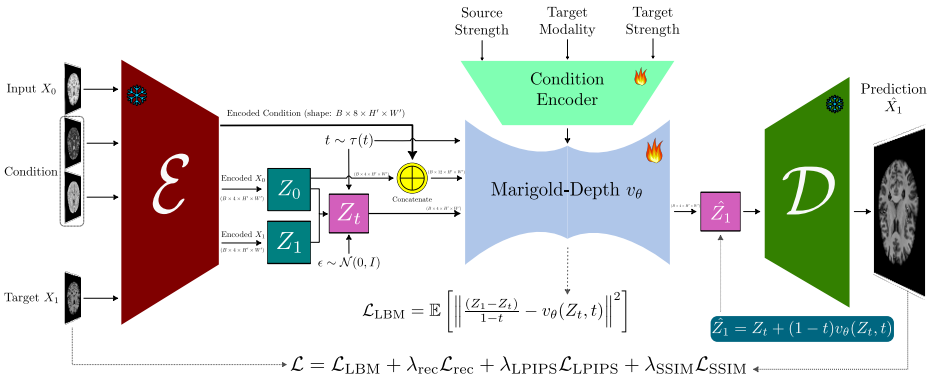


Fig. 2: Training conditional latent bridge matching model for MRI synthesis.

Bridge Model. The drift function v_θ is initialised from the Marigold-Depth [3] model, specifically the `prs-eth/marigold-depth-v1-1` checkpoint; Marigold is itself based on the Stable Diffusion (SD) v2 [10] UNet [11] and adapted to perform image-to-image translation for various image analysis tasks [3]. We extend Marigold's input convolutional layer to accept a 16-channel latent, corresponding to four latent groups of four channels each: the input image, the target image, and the two auxiliary modality condition slices, shown in Figure 2. We initialise the expanded convolution and replicate the weights of the first four input channels threefold and rescale by a factor of 1/3; the weights of the remaining four channels are copied without modification. This initialisation preserves the network's response at the start of training and is consistent with the initialisation used in Marigold [3]. v_θ is optimised end-to-end using the objective

$$\mathcal{L} = \mathcal{L}_{\text{LBM}} + \lambda_{\text{rec}}\mathcal{L}_{\text{rec}} + \lambda_{\text{LPIPS}}\mathcal{L}_{\text{LPIPS}} + \lambda_{\text{SSIM}}\mathcal{L}_{\text{SSIM}} \tag{10}$$

where $\mathcal{L}_{\text{LBM}}$ is the LBM loss defined in eq. (8), $\mathcal{L}_{\text{rec}}$ is the reconstruction loss, $\mathcal{L}_{\text{LPIPS}}$ is the LPIPS [14] loss, and $\mathcal{L}_{\text{SSIM}}$ is the structural similarity index measure (SSIM) [13] loss (defined as $1 - \text{SSIM}$).

Condition Encoder. The Stable Diffusion UNet backbone of Marigold is a text-to-image model and hence utilises additional conditioning inputs to guide the image generation process using cross-attention. We leverage this setup and design a lightweight condition encoder to encode the source strength (i.e. the field strength of the input X_0), the target modality and the target strength to guide the cross-strength translation. The source and target strengths, $s^{\text{src}}, s^{\text{tgt}}$, are first normalised to $[0, 1]$ by the maximum field strength in the dataset and mapped to fixed sinusoidal encodings

$$\gamma(s) = [\sin(s \cdot \omega_k); \cos(s \cdot \omega_k)]_{k=0}^{K-1} \in \mathbb{R}^{2K}, \omega_k = \frac{2\pi k}{K-1}$$

with $K = 8$. The resulting vectors $\gamma(s^{\text{src}})$ and $\gamma(s^{\text{tgt}})$ are then independently projected to cross-attention space ($\mathbb{R}^{1024}$ in the case of SD 2.1) through a learnt

linear map. The target modality is projected to the same cross-attention space using a learnt embedding table. The three embeddings are stacked to form the condition embedding, which is appended to the cross-attention context of v_θ.

3.2 Training

VAE Pretraining. We first train a variational autoencoder on the large unpaired dataset to encode and decode samples to latent space, following the paradigm of latent diffusion [10]. The loss combines the reconstruction, KL, and perceptual terms from eq. (9): $\lambda_{\text{rec}} = 0.5$ with $\mathcal{L}_{\text{rec}} = L^1$, $\lambda_{\text{KL}} = 1 \times 10^{-6}$, and $\lambda_{\text{LPIPS}} = 0.1$, where $\mathcal{L}_{\text{LPIPS}}$ uses the VGG network [12] applied patch-wise with patch size 64 and stride 64. We train for 25,000 optimiser steps with AdamW [8], batch size 4, and 4 gradient accumulation steps, using 2,500 linear warm-up steps to a peak learning rate of 3×10^{-5}. Weights are tracked with an exponential moving average (EMA) ($\beta = 0.9997$). We train on 3 NVIDIA A10 GPUs for a total of 28 hours.

Drift Model. We train the drift model and condition encoder on the smaller paired dataset using the bridge matching loss from eq. (10) with $\sigma = 0.001$ and the weighted time schedule $\tau(t) = 0.9\delta_{t=0} + 0.025\delta_{t=0.25} + 0.05\delta_{t=0.5} + 0.025\delta_{t=0.75}$. The loss weights nRMSE reconstruction ($\lambda_{\text{rec}} = 0.1$), SSIM ($\lambda_{\text{SSIM}} = 0.1$), and AlexNet [5] LPIPS [14] ($\lambda_{\text{LPIPS}} = 0.2$). The pipeline trains end-to-end with AdamW [8], batch size 2 and 16 gradient accumulation steps, over 500 linear warm-up steps followed by 10,000 training steps. The drift model v_θ uses a learning rate of 3×10^{-5}; the condition encoder uses 1.5×10^{-4}. We apply ZeRO-1 [9], mixed precision, and an EMA of the weights ($\beta = 0.9995$). Our model trains on 3 NVIDIA A10 GPUs for 40 hours.

3.3 Inference

At inference we are given three source contrasts (T1, T2, and T2FLAIR) of a subject, all imaged at field strength s^{src}, and we wish to generate a volume of the same subject at target modality m^{tgt} and target strength s^{tgt}. We generate each slice independently along the axial axis by first centre-cropping the three source contrasts to 384×384 and encoding each to latent space with the VAE encoder, producing three 4-channel latents that are concatenated to form the 12-channel source tensor Z_{src}. v_θ operates on a 16-channel input formed by concatenating this source tensor with the bridge state Z_t from eq. (3); however, since Z_1 is unavailable at inference (we wish to predict this target latent), we simply initialise the bridge state at Z_0, i.e. $Z_t := Z_0$. The triplet $(s^{\text{src}}, m^{\text{tgt}}, s^{\text{tgt}})$ is mapped by the condition encoder to a condition embedding, which together with Z_{src} is fixed throughout integration. The bridge integration proceeds as per section 2.2 on $[Z_{\text{src}}; Z_t]$; after N steps, the predicted target latent $\hat{Z}_1$ is mapped back to pixel space using the VAE decoder, and finally zero-padded back to the original resolution.

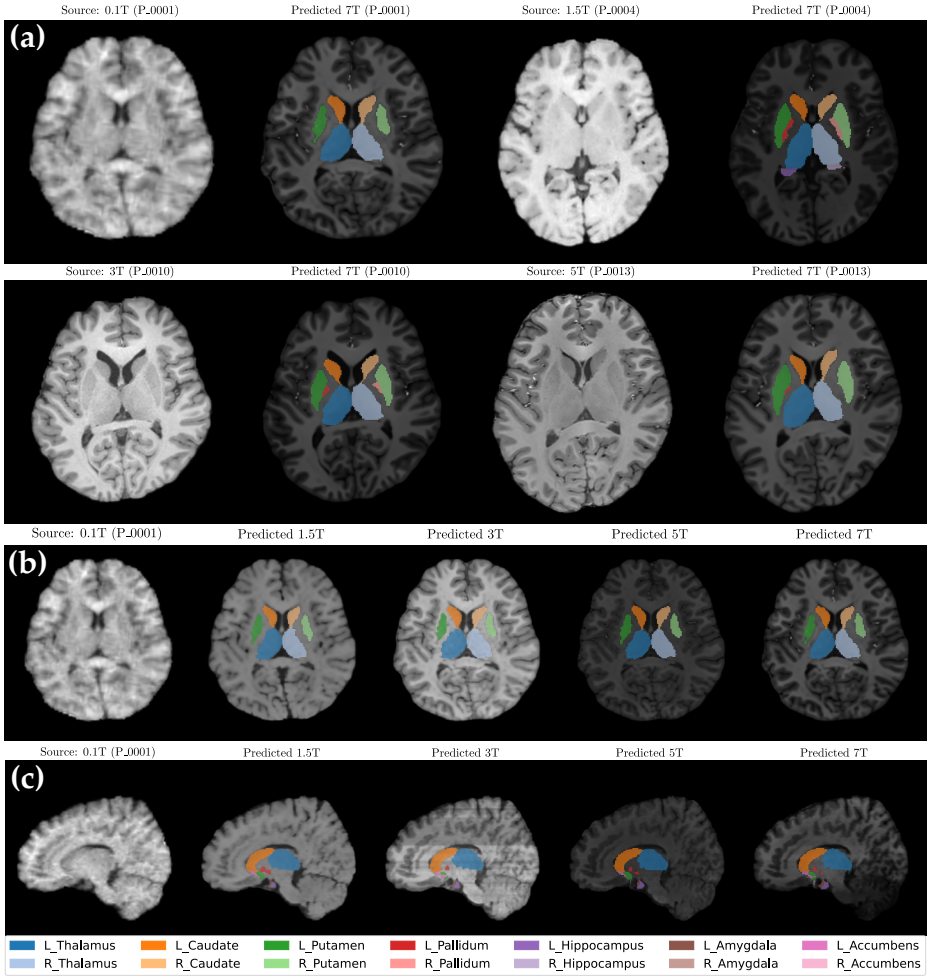
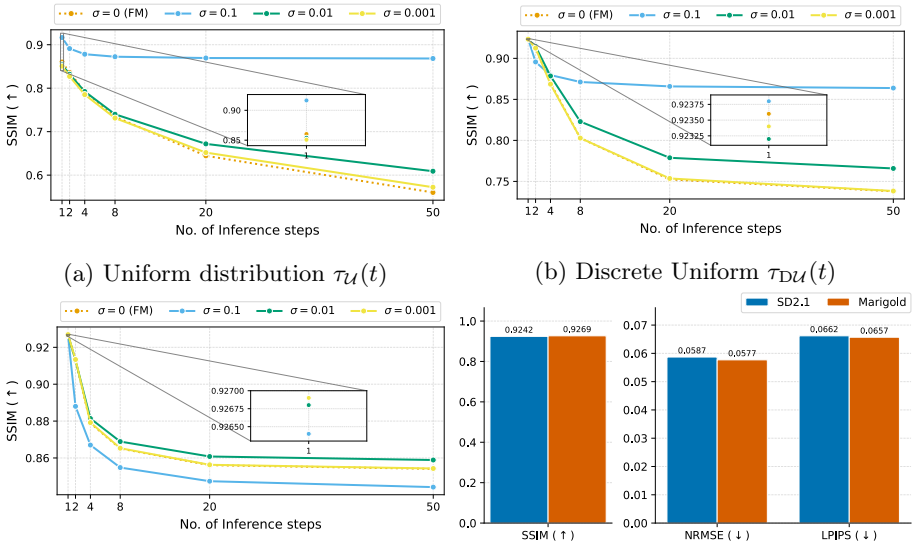


Fig. 3: Results for Task 1 and Task 2 using our single unified model. **(a)** T1W model outputs and deep grey matter (DGM) segmentations for ultra-high-field synthesis from low-field 2D slices (Task 1, $z = 160$). **(b)** Axial T1W model outputs and DGM segmentations for higher-field image synthesis from ultra-low-field slices (Task 2, $z = 160$). **(c)** Sagittal T1W outputs and DGM segmentations for higher-field image synthesis from ultra-low-field slices (Task 2, $z = 160$).

4 Results

We present our validation results for all three tasks of the MRIxFields2026 challenge in Table 1. Our unified model is competitive across all metrics on all three tasks on the validation leaderboard. We show outputs of our model and deep grey matter (DGM) segmentations in Figure 3. Figure 3a showcases ultra-high-field (7T) synthesis from low-field slices for four subjects on the validation



(a) Uniform distribution $\tau_{\mathcal{U}}(t)$

(b) Discrete Uniform $\tau_{\mathcal{DU}}(t)$

(c) Discrete Weighted $\tau_{\mathrm{DW}}(t)$

(d) Marigold vs SD2.1

Fig. 4: Ablation across σ, noise schedules $\tau(t)$, and number of inference steps.

dataset, overlaid with their corresponding SynthSeg segmentations [1]; Figure 3b showcases higher-field synthesis from a 0.1T volume for a single input subject. Our synthesis enhances detail and contrast in each of the slices with increasing field strength and preserves the size and location of DGM structures across all higher-field slices. Lastly, Figure 3c shows that since our method generates each 2D axial slice of a volume independently, inter-slice intensity consistency is reduced, shown as banding artefacts in sagittal sections. It is possible to reduce thse artefacts by generating volumes along the three spatial orientations and averaging; this improves every metric (e.g. lowers nRMSE from 0.289 to 0.266 on Task 1, 0.202 to 0.188 on Task 2, 0.230 to 0.219 on Task 3) except LPIPS (0.072 to 0.077 on Task 1, 0.092 to 0.103 on Task 2, 0.081 to 0.083 on Task 3) at the cost of triple the inference time per volume.

4.1 Ablations

For our ablations, we train all models for $5,000$ steps with 250 linear warm-up steps to a peak learning rate of 3×10^{-5} and track the weights with an EMA ($\beta = 0.9995$) using the Marigold UNet. We use our pretrained VAE for all experiments and only focus on the drift model v_θ; without loss of generality, we train on

Table 1: Validation metrics.

Task	SSIM	LPIPS	nRMSE	Dice	Volume
Task 1	0.914	0.072	0.289	0.898	0.880
Task 2	0.885	0.092	0.202	0.857	0.839
Task 3	0.902	0.081	0.230	-	-

2 out of 3 volunteers in the paired dataset and validate on the held-out volunteer. Our best model uses the Marigold UNet with $\sigma = 10^{-3}$ and the discrete-weighted schedule, achieving SSIM = 0.9269 in 1 inference step.

Impact of timestep distribution $\tau(t)$. We test three choices of the timestep distribution $\tau(t)$: a uniform distribution $\tau_{\mathcal{U}}(t) = \mathcal{U}([0, 1))$, a discrete-weighted distribution $\tau_{\mathrm{DW}}(t) = 0.9\delta_{t=0} + 0.025\delta_{t=0.25} + 0.05\delta_{t=0.5} + 0.025\delta_{t=0.75}$ (from the original LBM paper [2]), and lastly a discrete-uniform distribution $\tau_{\mathrm{D}\mathcal{U}}(t) = \mathcal{U}(\{0, 0.25, 0.5, 0.75\})$. The results are shown in Figures 4a to 4c.

Impact of bridge noise σ. We ablate values for the bridge noise σ in eq. (7): we test $\sigma = 0, 0.1, 0.01, 0.001$. $\sigma = 0$ is the flow-matching (FM) instantiation. The results are shown in Figures 4a to 4c.

Impact of model v_θ. We test the impact of two different choices of v_θ: the original SD 2.1 UNet from the `sd-research/stable-diffusion-2-1-base` checkpoint, as well as the Marigold-Depth model (Figure 4d). The SD 2.1 UNet has its `conv_in` weights replicated 4 times and rescaled by 1/4 to accept a 16-channel latent.

5 Conclusion

We present a single, unified model addressing all three tasks of the MRIx-Fields2026 challenge, based on conditional latent bridge matching. We can improve our method by generating volumes along three spatial orientations and averaging, producing improved distortion metrics at the expense of perceptual metrics, and at the cost of higher inference time.

Acknowledgments. The authors acknowledge the use of the Batch Compute System in the Department of Computer Science at the University of Warwick, and associated support services, in the completion of this work. Calculations were performed using the Sulis Tier 2 HPC platform hosted by the Scientific Computing Research Technology Platform at the University of Warwick. Sulis is funded by EPSRC Grant EP/T022108/1 and the HPC Midlands+ consortium. Computing facilities were provided by the Scientific Computing Research Technology Platform of the University of Warwick. S.S. is funded by the Department of Computer Science, University of Warwick. T.B. is supported through EPSRC/NSF grant EP/X026663/1.

References

1. Billot, B., Greve, D.N., Puonti, O., Thielscher, A., Van Leemput, K., Fischl, B., Dalca, A.V., Iglesias, J.E.: Synthseg: Segmentation of brain mri scans of any contrast and resolution without retraining. Medical Image Analysis **86**, 102789 (2023)
2. Chadebec, C., Tasar, O., Sreetharan, S., Aubin, B.: LBM: Latent Bridge Matching for Fast Image-to-Image Translation. In: International Conference on Computer Vision, ICCV 2025 (2025)

3. Ke, B., Qu, K., Wang, T., Metzger, N., Huang, S., Li, B., Obukhov, A., Schindler, K.: Marigold: Affordable adaptation of diffusion-based image generators for image analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–18 (2025)
4. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: International Conference on Learning Representations (ICLR 2014) (2014)
5. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proceedings of the 26th International Conference on Neural Information Processing Systems (2012)
6. Kullback, S., Leibler, R.A.: On Information and Sufficiency. The Annals of Mathematical Statistics **22**(1), 79 – 86 (1951)
7. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. In: International Conference on Learning Representations, ICLR 2022 (2022)
8. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (ICLR 2019) (2019)
9. Rasley, J., Rajbhandari, S., Ruwase, O., He, Y.: Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. p. 3505–3506. KDD '20, Association for Computing Machinery (2020)
10. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-Resolution Image Synthesis with Latent Diffusion Models. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022 (2022)
11. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. pp. 234–241 (2015)
12. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (ICLR 2015) (2015)
13. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E.: Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004)
14. Zhang, R., Isola, P., Efros, A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings - 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2018. pp. 586–595. IEEE Computer Society (2018)