Formulation Diversity and Boundary-Aware Postprocessing for Mitral Valve Segmentation in 3D TEE and 2D Surgical Video

Jesús David González Riveros[1] and Juan Pablo Betancur Rengifo[1]

Vall d'Hebron Institut de Recerca (VHIR), Barcelona, Spain
jesus.gonzalez.riveros@vhir.org
juan.betancur@vhir.org

Abstract. The MVAA 2026 challenge asks teams to segment the mitral valve in three kinds of images: cardiac CT (Task 1), 3D heart ultrasound (Task 2) and video from heart surgery (Task 3). We describe the models we submitted and how they scored on the final test set. A Dice overlap score of 0.839 on CT, 0.851 on ultrasound and 0.799 on video. Our main result is a clear negative finding on Task 2. The valve has two flaps (leaflets) that sit inside one shared outline. A network learns that shared outline well (0.89 Dice). We tried five ways to use it to tell the two flaps apart, and none of them helped. To find out whether the idea itself was wrong or just our predicted outline, we ran the same test again using the true outline from the labels, which we would not have at test time. That version improved the score by 0.049 Dice. So the idea works; our predicted outline was simply not accurate enough. We also report two practical lessons. First, how you split data for validation is important as learned on Task 3, testing on frames from surgeries the model had already seen gave a misleadingly high 0.960 Dice. Second, the same cleanup step can help or hurt depending on the image: removing small stray predictions lowered boundary error on CT but did nothing on video.

Keywords: Mitral valve segmentation · Cardiac CT · Transesophageal echocardiography · Surgical video · Semi-supervised learning · nnU-Net

1 Introduction

The mitral valve controls blood flow inside the heart. Outlining it accurately in medical images helps surgeons plan and carry out repairs. Different images serve different steps: cardiac CT is used before surgery, 3D ultrasound (called TEE) is used to measure the valve and its flaps, and video from the surgeon's camera could give feedback during the operation. The MVAA 2026 challenge covers all three in three tasks, and we entered all of them. Because the tasks share the same target but use very different images, they are a good way to see which methods carry over from one setting to another and which do not.

We use a few standard scores throughout. Dice Score (also written DSC) measures how much the predicted region overlaps the true region, from 0 (no

overlap) to 1 (perfect). Hausdorff distance (HD) is the single worst gap between the predicted boundary and the true boundary. Average surface distance (ASD) is the typical boundary gap. For Dice, higher is better; for HD and ASD, lower is better.

The three tasks. The datasets and task definitions follow the official MVAA challenge description [1]. **Task 1** is CT. We are given 27 labelled scans and about 1,040 scans without labels, and must produce a single valve mask per scan. **Task 2** is 3D ultrasound. We are given 105 labelled scans in which each voxel is marked as background, posterior flap (PML) or anterior flap (AML). The challenge scores each flap separately and averages the two; it does not score the combined whole-valve region. This detail matters for our main result (Sec. 3.2). **Task 3** is surgical video. We are given 1,559 frames, of which only 118 contain a labelled valve, plus many unlabelled frames we are allowed to use. The score combines overlap and boundary quality, with 40% of the weight on the two boundary terms:

$$\mathcal{M} = 0.6 \cdot \mathrm{Dice} + 0.2 \cdot \frac{1}{1 + \mathrm{HD}/20} + 0.2 \cdot \frac{1}{1 + \mathrm{ASD}/3}. \tag{1}$$

Figure 1 shows a sample of the nature of the data for each task.

What this paper contributes.

1. **A clear negative result (our main contribution).** On Task 2 a network learns the whole-valve outline well, but using that outline to separate the two flaps does not help, no matter which of five methods we use. A control test using the true outline shows the idea is sound and the problem is only the accuracy of the predicted outline (Sec. 3.2).
2. **Data splitting is a correctness issue, not a preference.** On Task 3, scoring a model on frames from surgeries it had trained on gave 0.960 Dice, far above its honest score. A subtler mistake, keeping two surgeries for validation while calling it leave-one-out, hid the method that finally won.
3. **The same cleanup step helps on CT and hurts on video.** Removing small stray predictions cut the worst boundary errors on CT, where those errors are stray blobs, but did nothing on video, where the error is on the edge of the main region (Fig. 2).
4. **Honest numbers on "diversity".** Combining two network designs gave only a tiny gain that is within normal fold-to-fold variation; combining two label formats gave essentially nothing. We report the spread, not just a single number.

Related work. We build on nnU-Net [2], a widely used, self-configuring framework for 3D medical segmentation on Tasks 1, and its larger residual-encoder version [3] on Task 2. For video we use a standard U-Net-style network [4, 5] from the `segmentation_models_pytorch` library [6]. Because most video frames

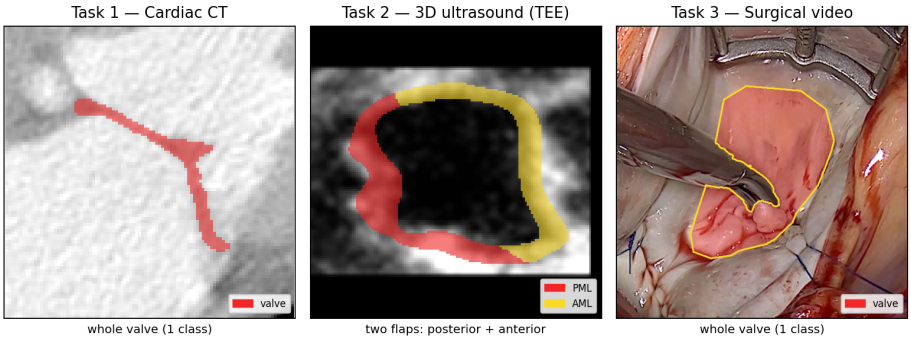

Fig. 1. What the training data looks like in each task, with the ground-truth. Left: a CT slice, where the valve is one region (red). Middle: a 3D US (TEE) slice, where the two flaps are labelled separately, posterior (PML, red) and anterior (AML, yellow). Right: a frame from surgery, where the valve is one region (red fill, yellow outline).

are unlabelled, we use semi-supervised learning, where the model helps label its own data; Mean Teacher [7] and FixMatch [8] are common choices and the challenge baseline uses the former. We also tested large "foundation" models, SAM 2 [9] and MedSAM [10], as a way to label frames automatically (Sec. 3.3). Boundary-aware losses [11] are a natural next step and we return to them at the end.

2 Methods

2.1 Shared setup

All experiments ran on two RTX A5000 GPUs. Tasks 1 and 2 use nnU-Net [2]. Task 3 uses PyTorch with the `segmentation_models_pytorch` library [6], starting from the official challenge code so we did not have to rewrite the semi-supervised parts. In Task 1, the unlabelled CT volumes are used online during training with Mean Teacher [7], while connected-component postprocessing is applied only to the final predictions. Tasks 2 and 3 reuse a small set of cleanup steps (keep the largest connected region, drop small stray regions, smooth the edge) and make careful use of the unlabelled scans by keeping only high-confidence automatic labels. How useful each of these is turns out to differ a lot between tasks, which is part of what we report.

2.2 Task 1: cardiac CT

The 27 labelled CT volumes differ in size and voxel spacing, so we use nnU-Net's standard preprocessing to resample the images to a common target spacing and normalise the CT intensities. The final `3d_fullres` configuration uses a standard PlainConvUNet with a patch size of $112 \times 128 \times 160$ voxels and a batch size of two.

We use three deterministic case-disjoint folds, with 18 labelled cases for training and nine for validation in each fold. In addition, the implementation excluded unlabelled volumes whose filename-derived numerical identifier matched a held-out labelled case identifier, leaving 1,031 of the 1,040 unlabelled volumes eligible in each fold. As no cross-subset patient metadata were available, these identifier matches cannot be interpreted as confirmed matches between biological patients.

To use the unlabelled data, we train the three models with Mean Teacher [7]. Training starts with 40 supervised-only epochs using a weighted combination of Dice and cross-entropy losses. From epoch 41 onward, the teacher generates hard voxel-level pseudo-labels online for the unlabelled patches. A voxel is included in the consistency loss when the maximum predicted class probability is at least 0.6. The teacher weights are updated as an exponential moving average of the student weights with a decay of 0.99, while the student receives additional Gaussian noise and voxel dropout. The contribution of the consistency loss is gradually increased during the first 30 semi-supervised epochs. The pseudo-labels are generated during training and are not stored or selected as a separate pseudo-labelled dataset.

For the final prediction, we use the student `checkpoint_final.pth` from folds 0, 1, and 2. Each model uses nnU-Net sliding-window inference with Gaussian weighting and mirroring along all three spatial axes. The three outputs are combined by equally averaging their pre-softmax logits. After restoring the prediction to the original image geometry, we keep only the largest 26-connected foreground component. This postprocessing is applied only to the final prediction and is not used during Mean Teacher training.

2.3 Task 2: 3D ultrasound

Data, models, and what we submitted. The 105 scans have uneven spacing and are mostly empty: about 44% of each scan is zero, the black fan shape outside the ultrasound cone. We use nnU-Net's standard preprocessing with no custom tricks. We trained two model designs on the same folds: the default nnU-Net, and a larger residual-encoder version (ResEnc-M). Our best result in internal testing came from averaging the two, but the model we actually submitted to the challenge is a single ResEnc-M model, which was simpler to package. This does not affect our challenge rank, because Task 2 counts the same for every eligible team, and on the hidden test the single model scored 0.8514 Dice (Table 1).

The idea we tested and dropped. The valve has two flaps that meet along a line (the coaptation line). A network can predict the whole valve, or each flap. We wondered whether predicting the whole valve first, then splitting it into the two flaps, would work better than predicting the flaps directly. We call using the whole-valve prediction to restrict the flaps "masking": anything outside the predicted whole valve is set to background. We tried this in five ways (listed in Table 3): masking within one model, masking across models, two ways of growing the flaps outward from confident seed points, and feeding the whole-

valve prediction back in as an extra input. We also tried simply averaging the whole-valve and per-flap models together. Section 3.2 reports what happened.

2.4 Task 3: surgical video

Data and how we validated. We read the labels from the image files the organisers provide, not from the polygon files, which they note are incomplete, and we drop one frame the organisers flagged as mislabelled. When the valve is visible it is large (about 9% of the frame), so this is not a "find the small object" problem. Of the 180 labelled frames, 62 have no valve and 61 show a prosthetic (artificial) valve but not the real one. Those prosthetic frames look like a valve but should be left blank, so they are the most useful negative examples, and we keep them in training as blank targets. We always validate by holding out whole surgeries, never single frames, because frames from the same surgery look almost identical and would leak information. We learned this the hard way (Sec. 3.3): a split we first called "leave-one-surgery-out" actually held out two surgeries, and the correct version holds out one at a time (six folds).

Delivered model For most of the challenge our best model was a plain valve-vs-background network (UNet++ with an EfficientNet-B5 backbone) trained with the Mean Teacher semi-supervised method and averaged across folds. The model we finally submitted, and that scored best on the hidden test, is different. It predicts three classes instead of two: background, surgical instruments, and valve. The extra "instruments" class helps the model on the hardest surgery, where tools are easily mistaken for tissue. The recipe is: (1) train this three-class model across the six folds; (2) use the averaged model to label the 1,379 unlabelled frames automatically, keeping only confident, stable predictions (about 993 of them pass); (3) retrain, counting the real labels eight times as heavily as the automatic ones so noise in the automatic labels cannot dominate; (4) at test time, average the six models and keep pixels the ensemble is at least 50% sure are valve, with simple flip-based averaging. The "keep only confident labels" rule is the same one we use on Task 1.

3 Experiments and Results

3.1 Task 1: cardiac CT

The three folds were trained independently using the same Mean Teacher configuration. Folds 0 and 1 completed the planned 200 epochs, while fold 2 stopped after 124 epochs according to the early-stopping criterion. Although the model with the best intermediate validation performance was also saved during training, the final student model from each fold was used for inference. The models from folds 0, 1, and 2 were then retained for the final ensemble.

Table 2 summarises the out-of-fold evaluation on the 27 labelled cases. Each labelled case was evaluated only by the fold in which it was held out from supervised training. Performance was relatively consistent across folds, although

Table 1. Scores of our submitted models on the hidden test set. Boundary errors (HD, ASD) are in millimetres for CT and ultrasound, and in pixels for video. Task 2 counts equally for all eligible teams, so it does not change the ranking.

Task	Images	Submitted model	Dice	HD	ASD
1	Cardiac CT	nnU-Net, 3-model average	0.8386	6.30	0.527
2	3D ultrasound	ResEnc-M, single model	0.8514	9.84	0.536
3	Surgical video	3-class average + auto-labels	0.7999	254.8	138.3

Table 2. Task 1 three-fold out-of-fold results after largest-component postprocessing. Surface distances are reported in millimetres.

Fold	Cases	Dice	ASD	HD95	HD
0	9	0.8551	0.2980	1.5990	4.8817
1	9	0.8593	0.2448	1.0201	3.7078
2	9	0.8421	0.2908	1.3284	4.7417
Overall	27	0.8522	0.2779	1.3159	4.4437

fold 2 obtained the lowest Dice. After applying the final largest-component postprocessing, the overall evaluation reached a mean Dice of 0.8522, an ASD of 0.2779 mm, an HD95 of 1.3159 mm, and a full Hausdorff distance of 4.4437 mm.

We also analysed retrospectively the effect of the connected-component postprocessing on the saved out-of-fold predictions. Before postprocessing, the mean Dice was 0.8520, ASD was 0.2799 mm, HD95 was 1.3024 mm, and full HD was 4.7270 mm. Keeping only the largest 26-connected foreground component therefore had almost no effect on Dice and only a small effect on ASD. The clearest improvement was observed in full Hausdorff distance, while HD95 did not improve overall. This suggests that the postprocessing mainly removed small distant false-positive components that produced large maximum surface-distance errors in a limited number of cases, rather than correcting the general boundary of the main valve region.

For the official challenge validation set, the three final student models were combined by equally averaging their pre-softmax logits. This three-model ensemble was the best retained Task 1 configuration and obtained a Dice score of 0.8591, a Hausdorff distance of 4.5515 mm, and an ASD of 0.2751 mm. The official validation performance was close to the internal out-of-fold results and supported the use of this ensemble as the final Task 1 model.

On the final hidden test set, the submitted three-model ensemble obtained a Dice score of 0.8386, an HD of 6.2989 mm, and an ASD of 0.5268 mm (Table 1). Performance was lower than on both the internal out-of-fold evaluation and the official validation set, with the largest difference appearing in the boundary-distance metrics. This gap highlights the importance of reporting final hidden-test performance in addition to internal and validation results.

Table 3. Task 2 negative result (five-fold, per-flap Dice after cleanup). Five methods try to use a predicted whole-valve outline to separate the two flaps; none beats the baseline. The last row is not a real method: it uses the true outline from the labels to show the best the idea could possibly do.

Method	Dice
Baseline (average of two model designs)	0.8451
(a) Whole-valve format, one model design	0.8408
(b) Whole-valve format, two model designs	0.8448
(c) Average of per-flap and whole-valve formats	0.8454
(d) Mask the flaps with the outline, one model	0.8401
(e) Mask the flaps with the outline, across models	0.8419
(f) Grow flaps from seeds (watershed)	0.8405
(g) Grow flaps from seeds (distance)	0.8402
(h) Feed the outline back in as an extra input	0.8325
Best possible: mask with the true outline	*0.8942*

3.2 Task 2: what "diversity" is really worth

Across five folds the default model scores 0.8409 Dice and the larger model scores 0.8422. The two are almost identical: the fold-by-fold difference between them is +0.0014, and its uncertainty range (95% confidence: −0.003 to +0.006) includes zero, so we cannot call it a real difference. For context, the folds themselves range from 0.833 (hardest) to 0.850 (easiest). Averaging the two models gives 0.8451, which is +0.004 over the single default model, but that gain is smaller than the fold-to-fold spread, so we do not claim it is statistically real. It does, however, point the same way under both label formats we tried, and the single model we submitted scored 0.8514 on the hidden test (Table 1). Averaging the two label formats added only +0.0007 (Table 3), which is nothing: at this data size the two formats learn essentially the same thing.

The challenge scores each flap separately and ignores the combined whole-valve region. So a method that improves the whole-valve outline but not the split between the two flaps earns nothing. Our negative result below is tied to that choice. A different application, for example measuring total valve area, would value the strong whole-valve outline that this scoring throws away.

Why all five methods failed. The network predicts the whole valve well (0.89 Dice). The gap between that 0.89 and the roughly 0.83 per-flap score tells us the hard part is not finding the valve, it is deciding which flap each voxel belongs to, along the line where the flaps meet. That is exactly where a "outline first, split second" approach should help, and it has worked for us before on blood vessels. It does not work here. Hard masking (d, e) removes correct valve voxels faster than it removes wrong ones. Growing from seeds (f, g) goes wrong where the two flaps touch. Feeding the outline back in (h) just adds a noisy signal. The last row of Table 3 settles the question: if we mask with the true outline,

Table 4. Task 3: how our submitted model improved on the hidden test. Dice higher is better; HD and ASD (pixels) lower is better. The winner adds a third class (instruments) and confident automatic labels.

Model	Dice	HD	ASD
Organizer baseline	0.572	556	303
Valve-vs-background, labelled frames only	0.665	592	424
Valve-vs-background, model average (EfficientNet-B5)	0.728	507	364
plus correct six-fold split	0.734	534	410
Three-class average + auto-labels (submitted)	**0.799**	**255**	**138**
Leading teams (approx.)	∼0.85	∼160	∼30

the score jumps to 0.8942. So the method is fine, and the only problem is that our predicted outline is not precise enough right at the line where the two flaps meet. This is different from blood vessels, where the branches are far apart; here the two flaps touch, and that is the exact spot where the decision is hardest. The practical advice: before building an "outline first, split second" pipeline, measure both the best-possible version and the real version. If they are far apart, spend your effort making the outline more accurate, not on the splitting step.

3.3 Task 3: test scores are not eval scores, and splits matter

When we scored a model on frames from surgeries it had already trained on, it got 0.960 Dice, far above its honest score of about 0.775. Nearby frames from one surgery look almost the same, so mixing them across training and testing inflates the score. This is why we always hold out whole surgeries. A quieter version of the same mistake cost us more time: for weeks we compared methods on a split that kept two surgeries for testing while we called it "leave-one-out". On that split, a strict rule ("only keep a method if it beats the current best on the hardest fold") kept rejecting the three-class model. Once we switched to the correct split and finished training the full set of models, the three-class model survived and then won.

Practice scores did not predict the real test. On the practice (eval) platform our valve-vs-background model peaked near 0.81 Dice, but on the hidden test its boundary errors stayed high (Table 4). The three-class model with automatic labels is the only one that improved overlap and boundaries at the same time, cutting the worst boundary error (HD) roughly in half. A gap to the leading teams remains, almost all of it in the boundary scores.

Large pretrained models did not work out of the box. We tried SAM 2 [9] and MedSAM [10] to label frames automatically, but they only worked when we told them where the valve was, which defeats the purpose. The valve is large, faint, and changes shape, which is a hard case for these models. So we labelled frames with our own in-domain model instead.

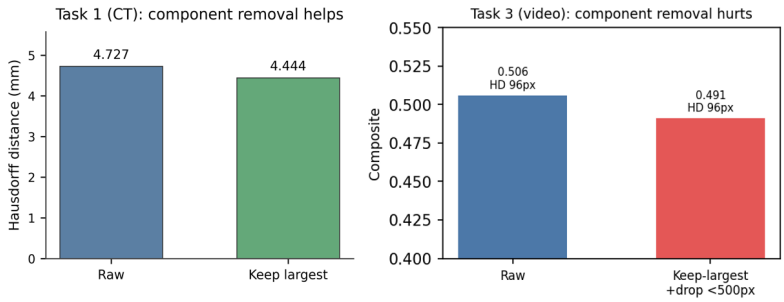

Fig. 2. The same cleanup step, opposite effect. On CT, one main connected region dominates, while small detached false positives can determine full HD; across 27 out-of-fold cases, keeping only the largest component reduced mean HD from 4.727 to 4.444 mm. On video the boundary error is on the edge of the main region, so deleting small blobs lowers the score (0.506 → 0.491) and leaves the worst boundary error unchanged (96.3 px).

4 Discussion and Conclusion

More supervision is not necessarily more useful supervision. Task 1 had far more unlabelled than labelled CT volumes, which made Mean Teacher a natural choice. However, about 99.96% of unlabelled voxels eventually passed the confidence threshold, making the filter almost non-selective. Since background strongly dominates the images, we also tried variants that increased the emphasis on foreground regions, but these did not improve validation performance. Without a supervised-only control we cannot isolate the contribution of Mean Teacher itself, but the experiment shows that more accepted pseudo-labels do not necessarily mean more informative supervision.

The Task 2 result is simple to state: the network learns the whole-valve outline well, but that does not help it separate the two flaps. Trying five different ways to use the outline did not change this. The control test with the true outline shows the idea is fine; our predicted outline is just not accurate enough where the two flaps meet. The lesson for others: check the best-possible version of an idea before spending time on it.

Tasks 1 and 3 both reward good boundaries, and both tempt you to clean up the prediction the same way. But the right fix is opposite (Fig. 2). On CT, keeping only the largest component barely changed Dice or ASD, but reduced full Hausdorff distance by removing a few distant false-positive regions; HD95 did not improve overall. On video the error is on the main region's edge, so deleting blobs does nothing; only a better model helped. A metric that rewards good boundaries does not tell you where the boundary error is. You have to look.

Task 1 used three folds for the final ensemble. Additional folds could have provided a more stable estimate and a larger ensemble, while part of our experimental effort was instead spent testing alternative training strategies, including foreground-emphasised variants that did not improve validation performance.

We also did not train a matched supervised-only baseline, so the individual contribution of Mean Teacher cannot be isolated from the final performance. Our Task 2 comparisons are on five folds whose natural spread is about as large as the differences we discuss, which is why we lean on the hidden-test scores rather than small point differences. On Task 3 a real gap to the leading teams remains, almost all in the boundary scores. No task used outside labelled data or special pretraining beyond ImageNet.

Conclusion. Across three kinds of images our submitted modelss scored 0.839 (CT), 0.851 (ultrasound) and 0.799 (video) on the hidden test. The lessons are simple and worth carrying to other problems: measure the best-possible version of an idea before building it; split validation data by whole patient or whole surgery, because practice scores did not predict the real test; and check where a boundary error actually is before cleaning it up, because the same step helped on CT and hurt on video.

References

1. Hammad, M., Bai, J., Zhao, J., Jia, T., Zhang, X., Xu, X., Ma, J., Li, S.: Mitral Valve Anatomy Analysis Using Multimodal Imaging Data. Zenodo, International Conference on Medical Image Computing and Computer Assisted Intervention 2026 (MICCAI) (2026). https://doi.org/10.5281/zenodo.19726755
2. Isensee, F., Jaeger, P.F., Kohl, S.A.A., Petersen, J., Maier-Hein, K.H.: nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. Nature Methods **18**(2), 203–211 (2021)
3. Isensee, F., Wald, T., Ulrich, C., et al.: nnU-Net revisited: a call for rigorous validation in 3D medical image segmentation. In: MICCAI 2024. LNCS, vol. 15009, pp. 488–498. Springer, Cham (2024)
4. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015)
5. Zhou, Z., Rahman Siddiquee, M.M., Tajbakhsh, N., Liang, J.: UNet++: a nested U-Net architecture for medical image segmentation. In: DLMIA. LNCS, vol. 11045, pp. 3–11. Springer, Cham (2018)
6. Iakubovskii, P.: Segmentation Models PyTorch. https://github.com/qubvel/segmentation_models.pytorch (2019)
7. Tarvainen, A., Valpola, H.: Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results. In: NeurIPS 30, pp. 1195–1204 (2017)
8. Sohn, K., Berthelot, D., Li, C.-L., et al.: FixMatch: simplifying semi-supervised learning with consistency and confidence. In: NeurIPS 33, pp. 596–608 (2020)
9. Ravi, N., Gabeur, V., Hu, Y.-T., et al.: SAM 2: Segment Anything in Images and Videos. arXiv:2408.00714 (2024)
10. Ma, J., He, Y., Li, F., Han, L., You, C., Wang, B.: Segment anything in medical images. Nature Communications **15**, 654 (2024)
11. Kervadec, H., Bouchtiba, J., Desrosiers, C., Granger, E., Dolz, J., Ben Ayed, I.: Boundary loss for highly unbalanced segmentation. Medical Image Analysis **67**, 101851 (2021)