From Patches to Global Representations: Context Window, Pooling, and VLA Distillation for CT

Tim Veenboer[1[0009−0004−5883−9500]], Razvan-Andrei Matisan[1[0009−0007−7548−4061]], George Yiasemis[1[0000−0002−1348−8987]], Jonas Teuwen[1[0000−0002−1825−1428]], and Kevin Groot Lipman[1[0000−0003−3651−2529]]

[1]Netherlands Cancer Institute (NKI), Plesmanlaan 121, 1066CX, Amsterdam, Netherlands
t.veenboer@nki.nl

Abstract. Foundation models for CT tend to provide either strong local representations or strong global representations. However, classification tasks in CT may require holistic decisions based on *needle-in-the-haystack* features within a scan, while in other cases they are driven by more localized evidence indicated by a foreground mask. This work presents a solution to 17 binary classification tasks in the linear-probing, all-data track of the CVPR 2026 Challenge: Foundation Models for General CT Image Diagnosis. Our method combines the local and global representations of two prominent CT foundation models, TAP-CT and SPECTRE, by distilling SPECTRE's crop and global `[CLS]` tokens into two intermediate stages after TAP-CT through alignment with a pooled representation of TAP-CT's patch grid. This intermediate representation is then used as a feature-extraction stage for probing, after which pooling strategies are applied to the frozen features. Averaged across the 17 tasks, the method achieves a validation AUC of 0.815 and a balanced accuracy of 0.804.

Keywords: CT foundation model · Linear probing · Embedding aggregation · Binary classification.

1 Introduction

Vision-only computed tomography (CT) classification using CT foundation model (FM) features faces several challenges. A key difficulty is that CT classification can often be framed as a *needle-in-the-haystack* problem. In such cases, the model must reach a holistic conclusion from the full set of scan features, even when the decisive abnormality (e.g. positive or negative for cancer) may occupy only a few voxels. In practice, however, the clinical context that motivated the scan is not available to vision-only CT FMs; the model is instead provided only with label supervision. In the present challenge setting, foreground masks are also available, which facilitate localization of abnormalities and reduce the reliance on purely global representations. This combination of tasks with and without foreground masks creates an opportunity to leverage both models that excel at local feature extraction and models that capture global context.

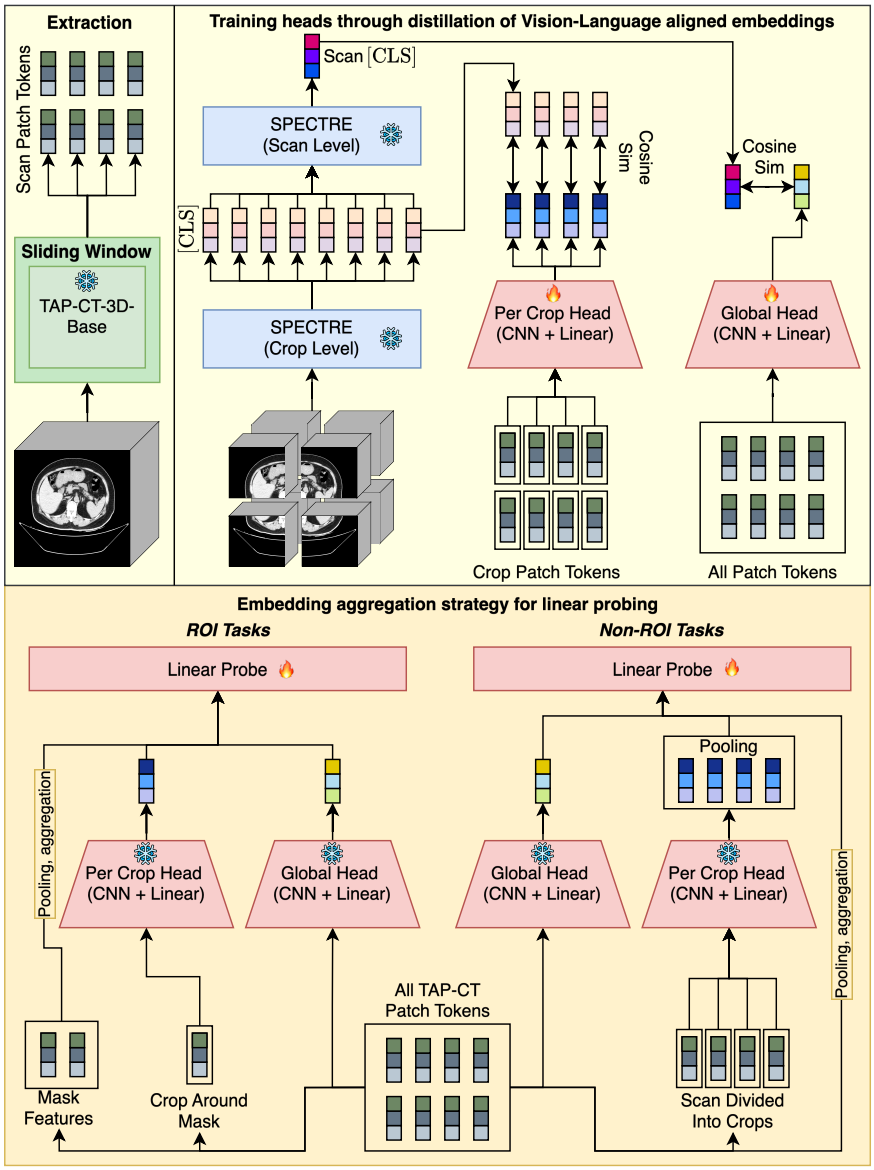


Fig. 1: Overview of the distillation procedure used to inject SPECTRE's global representations into intermediate stages of the final foundation model. Dedicated heads are used to distill both crop-level `[CLS]` tokens and the global vision–language-aligned `[CLS]` token from SPECTRE. For ROI tasks, the resulting features are combined with pooled foreground-mask patch features from TAP-CT, whereas for non-ROI tasks they are combined with pooled features computed from all TAP-CT patch tokens. For clarity of presentation, we omit TAP-CT's `[CLS]` tokens as input to the linear probe in this figure.

1.1 Related Work

The field of vision foundation models for CT has gradually shifted from Convolutional Neural Network (CNN)-based architectures [6,10,9,7] to Vision Transformer (ViT)-based solutions [4,12,3,2,14]. The level of feature granularity in these ViT-based CT foundation models varies for several reasons, including differences in resolution, spacing, crop size, and patch size.

Curia [3], 3DINO [14], SPECTRE [2], and TAP-CT [12] all employ variants of DINO-based pretraining [8,1,11]. DINOv2 and DINOv3 emphasize the joint optimization of local and global feature quality through the iBOT [17] and DINO objectives. For CT, however, the scale at which these objectives are applied is crucial. For notational clarity, this work uses the convention of $(\text{depth}, \text{height}, \text{width})$ for voxel space and (z, y, x) for natural coordinates. TAP-CT's 3D base model operates on a relatively small window of $12 \times 224 \times 224$ and uses small patches of $4 \times 8 \times 8$ voxels, mirroring the patch sizes used in 2D ViTs such as the original DINO models, Curia, and other TAP-CT variants. This enables the model to capture features at a very fine granularity, but limits its ability to model global context, since the context window covers only a small portion of the full scan. In contrast, SPECTRE and 3DINO use larger patch sizes of $8 \times 16 \times 16$ and $16 \times 16 \times 16$, respectively, thereby packing 8 and 16 times as many voxels into a single patch as TAP-CT or Curia. This design allows them to operate on larger crop sizes and capture more global context within the scan. Together, these differences create an opportunity to combine models that are specialized for local detail with models that are better suited to global reasoning.

1.2 Contributions

This work addresses the 17 binary classification tasks posed by the *Foundation Models for General CT Image Diagnosis*[1] challenge. Of these, 11 tasks provide foreground ROI masks for the target condition, while the remaining 6 require classification over the full field of view. All tasks are evaluated using frozen features with a linear probe. To address both settings, we combine the complementary strengths of TAP-CT and SPECTRE within a unified model. Specifically, SPECTRE's global knowledge is distilled into an intermediate stage between TAP-CT and the probe. This work's contributions can be summarized as follows:

1. We propose a distillation framework that injects global knowledge from SPECTRE into TAP-CT by aligning pooled TAP-CT patch representations with SPECTRE's global CLS token.
2. We show that the classification performance of existing CT foundation models can be improved by training only an intermediate representation layer, rather than fine-tuning the underlying foundation model backbones.

[1] https://www.codabench.org/competitions/12650/

3. We demonstrate that combining local and global CT representations in this manner is effective for both ROI-based and non-ROI classification tasks in a linear-probing setting.

2 Method

This section describes the backbone architectures, the proposed framework, and the task-specific pooling mechanisms for ROI and non-ROI settings.

2.1 Backbone Architectures

SPECTRE and the 3D ViT-base variant of TAP-CT (TAP-B-3D) are used in the distillation phase. SPECTRE comprises two stages: an initial stage that processes non-overlapping crops of size $64 \times 128 \times 128$, each partitioned into patches of $8 \times 16 \times 16$ with embedding dimension $d = 1080$, and a second stage that acts as a scan-level feature combiner. For each crop, the patch tokens are pooled and merged with the corresponding `[CLS]` token to form a per-crop global representation. These representations are then aggregated by the second stage. During pretraining, the first stage is optimized with DINOv3, whereas the second stage is vision-language aligned (VLA) with Qwen 3 [16].

TAP-B-3D divides a scan into windows of size $12 \times 224 \times 224$ and operates with a patch size of $4 \times 8 \times 8$. The model is pretrained using DINOv2-based optimization. During inference, scans are processed with a sliding window of size $12 \times 224 \times 224$ and stride 4. Overlapping patch tokens are merged using Gaussian smoothing, while all `[CLS]` tokens are collected into a set without smoothing.

2.2 Distillation Framework

The goal of the distillation framework is to align TAP-CT's local feature representations with SPECTRE's crop-level `[CLS]` tokens from the first stage, as well as the global `[CLS]` token produced by the second-stage feature aggregation module. To this end, we employ two separate intermediate heads to extract and project representations from SPECTRE's two stages. The overall distillation pipeline is illustrated in Figure 1.

We now formalize the proposed distillation framework. Let e denote the frozen TAP-CT feature extractor. We introduce two lightweight CNN plus linear pooling modules: a crop-level pooling module f_c and a global pooling module f_g. Given a crop $\boldsymbol{x}_c$, the crop-level pooled representation is defined as

$$\boldsymbol{t}_c = f_c(e(\boldsymbol{x}_c)). \tag{1}$$

Similarly, given a full scan $\boldsymbol{x}$, the global pooled representation is defined as

$$\boldsymbol{t}_{all} = f_g(e(\boldsymbol{x})). \tag{2}$$

Let s_1 and s_2 denote the two stages of SPECTRE. The first stage produces a crop-level `[CLS]` token,

$$\boldsymbol{c} = s_1(\boldsymbol{x}_c), \tag{3}$$

while the second stage aggregates the set of crop-level representations from all non-overlapping crops C within the scan to produce a scan-level `[CLS]` token,

$$\boldsymbol{g} = s_2\big(\{s_1(\boldsymbol{x}_c)\}_{c \in C}\big)\,. \tag{4}$$

The crop-level pooling module is trained to align $\boldsymbol{t}_c$ with the corresponding SPECTRE crop representation $\boldsymbol{c}$ by maximizing their cosine similarity:

$$\mathcal{L}_{crop} = 1 - \frac{\boldsymbol{t}_c \cdot \boldsymbol{c}}{||\boldsymbol{t}_c|| \cdot ||\boldsymbol{c}||}. \tag{5}$$

Likewise, the global pooling module is trained to align $\boldsymbol{t}_{all}$ with SPECTRE's global representation $\boldsymbol{g}$:

$$\mathcal{L}_{global} = 1 - \frac{\boldsymbol{t}_{all} \cdot \boldsymbol{g}}{||\boldsymbol{t}_{all}|| \cdot ||\boldsymbol{g}||}. \tag{6}$$

Both objectives are optimized independently, while the TAP-CT and SPECTRE backbones remain frozen throughout training. Each intermediate stage consists of three convolutional layers interleaved with Group Normalization [13] and GELU activations [5]. These layers are followed by an adaptive pooling operation and a two-layer projection head with an intermediate GELU activation.

The crop-level intermediate stage uses a hidden dimensionality of 1024, whereas the global intermediate stage uses a hidden dimensionality of 384. These architectural choices were determined empirically through a hyperparameter sweep.

2.3 Pooling, aggregation and the probe

To maximize performance, we employ various aggregation and pooling strategies to combine the various embeddings obtained by the feature extractor (TAP-CT) and the two distilled intermediate stages at inference time. We can divide these strategies between an ROI and Non-ROI version. Table 1 provides an overview of the different pooling methods and their respective embedding sizes.

ROI: For tasks with an available foreground mask, we apply max, mean, and standard deviation pooling over TAP-CT features restricted to the masked region, yielding three feature vectors. The same three pooling operations are

Table 1: Overview of the different pooling and aggregation strategies applied to create an input vector for the linear probe.

ROI	*dim*	**Non-ROI**	*dim*
TAP-CT Features			
Masked mean pool	768	Global mean pool	768
Masked max pool	768	Global max pool	768
Masked std pool	768	Global std pool	768
Context ring pool	768	Top-K pool	768
Multi-scale views	3072	Depth-binned pool	3072
Global mean pool	768	—	—
`[CLS]` mean	768	`[CLS]` mean	768
`[CLS]` max	768	`[CLS]` max	768
`[CLS]` std	768	`[CLS]` std	768
HU statistics	16	HU statistics	16
Distillation features			
Global	1080	Global	1080
Per-crop (centered)	1080	Per-crop (mean)	1080
—	—	Per-crop (max)	1080
Total	**11392**	**Total**	**11704**

additionally applied to the set of TAP-CT `[CLS]` tokens obtained via a sliding window over the mask region, resulting in three more feature vectors.

Furthermore, we extract multi-scale contextual information from TAP-CT features by constructing four cubic regions centered at the mask centroid. For each scale, voxel embeddings within the corresponding cube are mean-pooled, yielding a single feature vector. Each resulting vector is then individually ℓ_2-normalized, resulting in four additional feature vectors.

In addition, we compute contextual features by mean-pooling embeddings from a region immediately surrounding the mask, capturing information just outside the foreground region. We also include a global feature obtained by mean-pooling all patch embeddings, independent of the mask. To incorporate basic intensity information, we append a small set of summary statistics computed from HU values within the mask. Finally, we include representations from both the global and crop-level intermediate distillation stages, where each crop is centered at the mask centroid.

Non-ROI: We apply max, mean, and standard deviation pooling over all available patch tokens and `[CLS]` tokens. Additionally, we partition the scan along the depth dimension into n bins ($n = 4$), and compute the mean within each bin, followed by ℓ_2-normalization, yielding four feature vectors. We further select the top-k patches based on their ℓ_2-norm ($k = 64$), and apply mean pooling over this subset to obtain an additional feature representation. A small set of summary statistics based on Hounsfield Unit (HU) intensities is also appended. Finally, we include the representation from the global intermediate stage. Since

no mask is available, the scan is partitioned into non-overlapping crops, which are all passed through the per-crop intermediate stage. The resulting crop-level feature vectors are aggregated using mean and max pooling.

3 Experiments

This section describes the datasets used, the evaluation metrics, preprocessing procedures, the experimental environment and training protocols, and finally the qualitative and quantitative results.

3.1 Dataset and evaluation metrics

The challenge was divided into multiple tasks and sub-tracks. This work addresses Task 1, Track 1: linear probing a pretrained CT foundation model on 17 binary classification tasks across three datasets. In this track, pretraining is performed using the full set of pretraining data made available within the challenge. The distillation process uses the full AMOS dataset, all scans from the FLARE dataset, and 3,232 volumes from the Merlin dataset. These volumes were derived through label stratification, we selected volumes that were relevant to disease conditions present in the challenge. The majority of classification tasks are highly imbalanced; therefore, we apply weighted random sampling to increase the influence of infrequent labels during training.

Evaluation metrics: The classification metrics used to assess each binary task are:

- **Balanced Accuracy (BalAcc)**: the average of sensitivity and specificity, robust to class imbalance.
- **AUROC**: the area under the receiver operating characteristic curve.

The efficiency metric is:

- **Running time (s)** per case, measured including Docker startup time. Running time is the average of feature-extraction time per target.

3.2 Implementation details

Preprocessing pipeline We pre-extract and store the TAP-CT patch grids, SPECTRE crop-level `[CLS]` tokens, and SPECTRE scan-level `[CLS]` tokens for each scan prior to distillation. This enables faster experimentation across different training configurations.

SPECTRE: We resample the volumes to 2mm $\times$ 1mm $\times$ 1mm rather than retaining the original 1.0mm $\times$ 0.5mm $\times$ 0.5mm resolution in order to accelerate feature extraction. This preserves the relative voxel aspect ratios. Moreover, SPECTRE was pretrained to be robust to resolution deviations and scale

changes. For each volume, HU intensities are clipped to $[-1000, 1000]$ and min–max normalized to the range $[0, 1]$. Orientation of the volumes is set to RAS.

TAP-CT: Volumes are resized to $z \times 224 \times 224$ in voxel space. The volumes are clipped and z-score normalized as described in [12]. All volumes are loaded in LPS coordinates.

To ensure spatial alignment between the extracted SPECTRE and TAP-CT regions, SPECTRE's output is mapped to LPS coordinates and matched to the corresponding image coordinates using a stored crop center from the SPECTRE extraction step.

Environment settings The development environments and requirements are presented in Table 2.

Table 2: Development environments and requirements. (mandatory table)

System	Ubuntu 22.04.5 LTS
CPU	32 × AMD EPYC 7542 32-Core Processor
RAM	224GB
GPU (number and type)	1 × NVIDIA A6000 48G (global) 2 × NVIDIA A100 80G (per-crop)
CUDA version	12.8
Programming language	Python 3.12.11
Deep learning framework	Torch 2.10.0, torchvision 0.25.0, xformers 0.0.35

Training protocols As we work with pre-extracted embedding grids, augmentations are applied only after feature extraction and only to the global grids. The per-crop TAP-CT grid samples must remain aligned with their corresponding SPECTRE counterparts, and applying augmentations at this stage could disrupt that correspondence. The augmentations applied to the global TAP-CT grids include random flips, dropout, and Gaussian noise.

To avoid sampling the full TAP-CT grid and the corresponding reduction in batch size to 1 caused by variation along the z-axis, we extract a fixed crop of $48 \times 28 \times 28$ from the TAP-CT grid as a global representation of the scan. This crop is then used for distillation against SPECTRE's combiner `[CLS]` token.

The intermediate heads are selected by a simple hyperparameter sweep over learning rate and hidden dimensionality. An overview of the final set of hyperparameters can be found in Table 3.

3.3 Quantitative results on the validation set

Tables 4 and 5 report the validation performance of TAP-CT, SPECTRE, and the distillation method proposed in this paper in terms of AUROC and Balanced

Table 3: Training protocols for Task 1 — Linear Probing (pretraining). †: Volume sizes are reported in the embedding space of TAP-CT-B-3D. Specifically, CT volumes are divided into patches of $4 \times 8 \times 8$, such that a volume of $48 \times 28 \times 28$ corresponds to an original input size of $192 \times 224 \times 224$.

	Global	Per-crop
Pretrained backbone (if any)	TAP-CT-B-3D & SPECTRE	
Backbone tuning strategy	Distillation	
Batch size (pretraining)	32	64
Volume size†	$48 \times 28 \times 28$	$16 \times 16 \times 16$
Total pretraining epochs	100	50
Optimizer	AdamW	
Initial learning rate (lr)	1e-3	
Lr decay schedule	CosineAnnealingLR	
Pretraining time	12 hours	43 hours
Loss function	Cosine Similarity	
Number of model parameters	16.5M	80M
Number of FLOPs	80.4 G/sample	25.8 G/sample

Accuracy, respectively. The TAP-CT and SPECTRE baselines also employ the pooling and aggregation strategies described in Section 2.3. The results show that the two models are complementary across several tasks. For instance, TAP-CT substantially outperforms SPECTRE on *Adrenal Hyperplasia* and *Splenomegaly*, whereas the opposite trend is observed for *Colorectal Cancer* and *Liver Cyst*. The proposed distillation approach exploits this complementarity and achieves validation performance that is closer to that of the best-performing model across tasks. COVID and LUNA25 are not evaluated for TAP-CT, as these tasks were introduced into the challenge at a later stage.

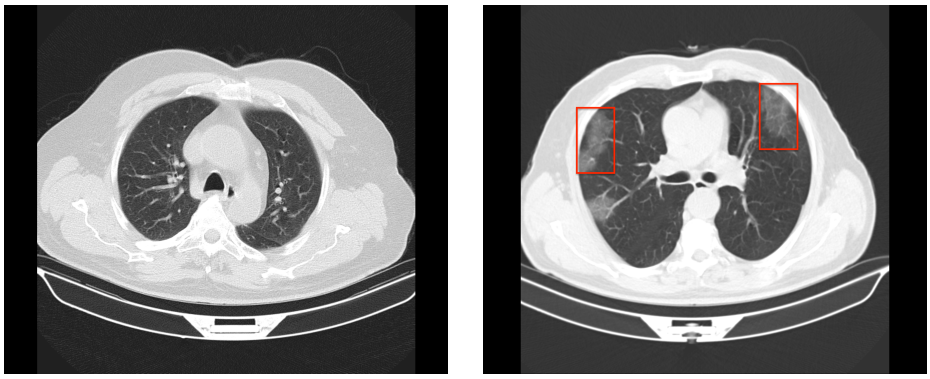
3.4 Qualitative analysis on the validation set

Figure 2 presents two examples from the COVID-CT dataset that were evaluated in this challenge. COVID-19 is often highly conspicuous in CT scans, which can make classification relatively straightforward for the model. This is reflected in the example on the right, which is correctly classified as COVID-19. In contrast, the example on the left is incorrectly predicted as positive despite being a negative case. In this patient, the presence of a particularly high concentration of bronchioles may have contributed to the misclassification.

Figure 3 shows another qualitative example from kidney stone classification on the AMOS dataset. The model correctly identifies a small kidney stone in the example on the right. In the example on the left, the kidney stone lies outside the segmentation mask. Since the linear probes for ROI tasks primarily use information from within the segmentation, apart from the context ring pooling and

Table 4: Per-condition AUROC for SPECTRE, TAP-CT, and our Distillation method. Best value in each row shown in bold green. – = not evaluated. $n = 17$ conditions.

	AUROC		
Condition	**SPECTRE**	**TAP-CT**	**Distillation**
Adrenal hyperplasia	0.638	0.766	**0.778**
Ascites	0.837	0.787	**0.868**
Atherosclerosis	0.772	0.772	**0.790**
Cholecystitis	0.750	0.702	**0.801**
Colorectal cancer	**0.879**	0.762	0.781
COVID	**0.975**	–	0.938
Fatty liver	0.861	**0.983**	0.965
Gallstone	0.846	**0.887**	0.841
Hydronephrosis	**0.940**	0.889	0.936
Kidney stone	0.714	0.714	**0.797**
Liver calcifications	0.802	0.824	**0.843**
Liver cyst	**0.691**	0.622	0.686
Liver lesion	**0.727**	0.713	0.722
Lung nodule malignancy (LUNA25)	**0.750**	–	0.662
Lymphadenopathy	0.725	0.780	**0.799**
Renal cyst	0.696	0.699	**0.715**
Splenomegaly	0.802	0.932	**0.940**
Average	0.788	0.789	**0.815**



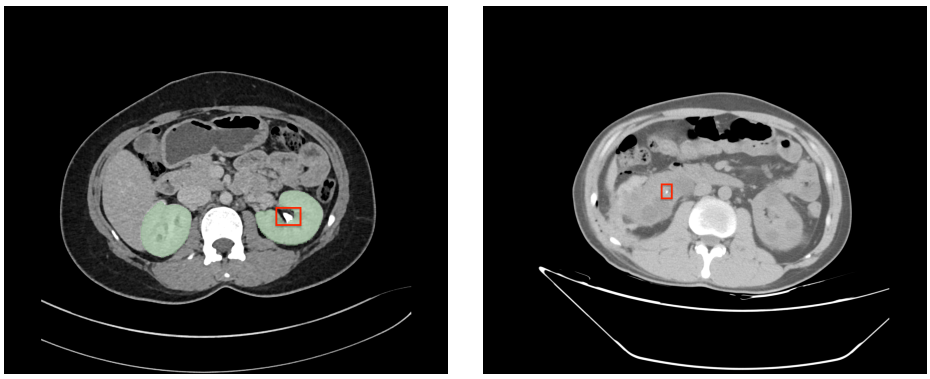
(a) Wrongly classified COVID-CT case (patient 367, false positive).

(b) Correctly classified COVID-CT case (patient 127, true positive).

Fig. 2: Qualitative examples of the proposed method on the COVID-CT dataset. The correctly classified case shows clear signs of lung infection associated with COVID-19.

Table 5: Per-condition balanced accuracy for SPECTRE, TAP-CT, and our Distillation method. Best value in each row shown in bold green. – = not evaluated. $n = 17$ conditions.

	Balanced Accuracy		
Condition	**SPECTRE**	**TAP-CT**	**Distillation**
Adrenal hyperplasia	0.700	0.771	**0.786**
Ascites	**0.853**	0.794	0.838
Atherosclerosis	0.765	**0.779**	0.750
Cholecystitis	0.745	0.726	**0.774**
Colorectal cancer	**0.846**	0.769	0.808
COVID	**0.969**	–	0.938
Fatty liver	0.846	**0.949**	**0.949**
Gallstone	**0.828**	0.819	0.819
Hydronephrosis	0.922	0.828	**0.938**
Kidney stone	0.715	0.715	**0.759**
Liver calcifications	0.792	0.819	**0.847**
Liver cyst	**0.676**	0.635	0.656
Liver lesion	**0.724**	0.702	0.716
Lung nodule malignancy (LUNA25)	**0.725**	–	0.675
Lymphadenopathy	0.778	0.750	**0.806**
Renal cyst	0.669	0.673	**0.696**
Splenomegaly	0.771	**0.929**	0.914
Average	0.784	0.777	**0.804**



(a) Wrongly classified kidney stone case (patient AMOS0289, false negative).

(b) Correctly classified kidney stone case (patient AMOS5310, true positive).

Fig. 3: Qualitative examples of the proposed method on kidney stone classification in the AMOS dataset. The correctly classified case shows a kidney stone in the right kidney. In the misclassified case, the kidney stone lies outside the segmentation of the patient's left kidney.

global mean pooling described in Section 2.3, the stone may not be sufficiently represented for the probe, potentially leading to misclassification.

3.5 Results on the final testing set

Table 6: Final test-set results for the *Foundation Models for General CT Image Diagnosis* challenge presented at the CVPR 2026 workshop. The table summarizes the performance of submitted solutions for Task 1, Track 1, which focuses on linear probing using frozen CT foundation model representations. Results are reported across the diverse undisclosed diagnostic tasks included in the challenge benchmark.

Rank	Team	Balanced Accuracy	AUROC	Runtime
1	fomofo	**0.654**	**0.714**	9.414
1	raidium	0.639	0.707	**4.393**
3	cemrg	0.643	0.705	4.467
4	maest	0.600	0.624	6.273
5	xupy	0.566	0.599	10.86
6	sdfmu	0.160	0.170	33.83

The final test-set results of the challenge, shown in Table 6, indicate that our solution (team fomofo) achieves the best performance in both Balanced Accuracy and AUROC. Its runtime was comfortably within the 40-second maximum. However, it is longer than runtimes of the other submissions, likely because it requires a forward pass through the TAP-CT backbone using a sliding window, as well as forward passes through both the per-crop and global intermediate stages.

4 Conclusion

We have demonstrated that distilling two CT foundation models, TAP-CT and SPECTRE, allows us to combine their complementary strengths into a single learned pipeline. The resulting model achieves consistently strong performance on both the validation and test sets of the CVPR 2026 workshop challenge, Foundation Models for General CT Image Diagnosis. These results suggest that a single model can effectively capture both local and global features. Moreover, they show that the strengths of multiple models can be combined without retraining either one.

4.1 Limitations and Future Work

Vision-only, volume-level CT classification benchmarks are inherently challenging. Disease-related signals are often confined to only a small number of voxels,

making them difficult to identify without prior knowledge. A representative example in this challenge is the LUNA25 dataset as posed here, where the task is to determine whether lung nodules spanning only a few voxels are benign or malignant. This is particularly difficult because, unlike the original challenge setup in which only a crop around the nodule was considered, the model here is provided only with a binary positive or negative label and is effectively expected to localize the relevant evidence on its own. In real-world settings, however, such prior knowledge is typically available: diagnostic scans are usually acquired only after clinical examination has identified plausible root causes of a patient's discomfort. Consequently, scans are rarely performed in the absence of symptoms. Although the method proposed in this work is able to disambiguate scans and demonstrates that there is signal available for classification, it remains an open problem to extract highly localized information and draw holistic conclusions in a vision-only setting.

Furthermore, although model combinations through distillation and ensembling can solve these tasks to a reasonably satisfactory extent, it would still be preferable to have a single model, pretrained once in an amortized fashion, that can represent both fine-grained structures and the conclusions derived from them. However, given the large and high-context search space inherent to CT, optimizing for both objectives within a single model remains difficult without incorporating supervisory signals from language.

Acknowledgements We thank all the data owners for making the medical images publicly available and CodaLab [15] for hosting the challenge platform. We further thank the Research High Performance Computing (RHPC) NKI for providing compute, and specifically maintainers Daniël Vis and Ameer Alkhier.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021) 3
2. Claessens, C., et al.: Scaling self-supervised and cross-modal pretraining for volumetric CT transformers. arXiv preprint arXiv:2511.17209 (2025) 3
3. Dancette, C., et al.: Curia: A multi-modal foundation model for radiology. arXiv preprint arXiv:2509.06830 (2025) 3
4. Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021) 3
5. Hendrycks, D., Gimpel, K.: Bridging nonlinearities and stochastic regularizers with gaussian error linear units. CoRR **abs/1606.08415** (2016), http://arxiv.org/abs/1606.08415 5

6. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998). https://doi.org/10.1109/5.726791 3
7. Li, W., Yuille, A., Zhou, Z.: How well do supervised models transfer to 3D image segmentation? In: The Twelfth International Conference on Learning Representations (2024) 3
8. Oquab, M., Darcet, T., Moutakanni, T., et al.: DINOv2: Learning robust visual features without supervision. Transactions on Machine Learning Research (2024) 3
9. Pai, S., Bontempi, D., Hadzic, I., Prudente, V., Sokač, M., Chaunzwa, T.L., Bernatz, S., Hosny, A., Mak, R.H., Birkbak, N.J., Aerts, H.J.W.L.: Foundation model for cancer imaging biomarkers. Nature Machine Intelligence **6**(3), 354–367 (Mar 2024). https://doi.org/10.1038/s42256-024-00807-9, https://doi.org/10.1038/s42256-024-00807-9 3
10. Pai, S., et al.: Vision foundation models for computed tomography. arXiv preprint arXiv:2501.09001 (2025) 3
11. Siméoni, O., Vo, H.V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., et al.: Dinov3. arXiv preprint arXiv:2508.10104 (2025) 3
12. Veenboer, T., Yiasemis, G., Marcus, E., Van Veldhuizen, V., Snoek, C.G.M., Teuwen, J., Groot Lipman, K.B.W.: Tap-ct: 3d task-agnostic pretraining of computed tomography foundation models. arXiv preprint arXiv:2512.00872 (2025) 3, 8
13. Wu, Y., He, K.: Group normalization. In: Proceedings of the European conference on computer vision (ECCV). pp. 3–19 (2018) 5
14. Xu, T., et al.: A generalizable 3D framework and model for self-supervised learning in medical imaging. npj Digital Medicine **8**(1), 639 (2025) 3
15. Xu, Z., Escalera, S., Pavao, A., et al.: Codabench: Flexible, easy-to-use, and reproducible meta-benchmark platform. Patterns **3**(7), 100543 (2022) 13
16. Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al.: Qwen3 technical report. arXiv preprint arXiv:2505.09388 (2025) 4
17. Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., Kong, T.: ibot: Image bert pre-training with online tokenizer. arXiv preprint arXiv:2111.07832 (2021) 3

Table 7: Checklist Table. Please fill out this checklist table in the answer column.
(**Delete this Table in the camera-ready submission**)

Requirements	Answer
A meaningful title	Yes
The number of authors ($\leq$6)	5
Author affiliations and ORCID	Yes
Corresponding author email is presented	Yes
Task and track (LP All / LP Coreset / EAO VO / EAO VL) are clearly stated in the abstract	Yes
Validation BalAcc and AUROC are presented in the abstract	Yes
Introduction includes at least three parts: background, related work, and motivation	Yes
A pipeline/network figure is provided	Figure number 1
Pre-processing	Page number 7
Strategies for data augmentation	Page number 8
Strategy for class imbalance	Page number 7
Strategies to improve model inference (under 40s/case)	N.A.
Post-processing	N.A.
Environment setting table is provided	Table number 2
Training protocol table is provided (Task 1 and/or Task 2)	Table number 3
Ablation study	Page number TBD
Efficiency evaluation results (running time) are provided	N.A.
Qualitative example (success and failure) is provided	Figure number 2 & 3
Limitation and future work are presented	Yes
Reference format is consistent	Yes
Main text $\geq$ 8 pages (not including references and appendix)	Yes