ce/papers Proceedings in civil engineering

Ernst & Sohn A Wiley Brand

ORIGINAL

EUROSTRUCT

Towards Workflows for the Use of AI Foundation Models in Visual Inspection Applications

Mattia Rigotti[1] | Diego Antognini[1] | Roy Assaf[1] | Kagan Bakirci[1] | Thomas Frick[1] | Ioana Giurgiu[1] | Klára Janoušková[1] | Filip Janicki[1] | Husam Jubran[1] | Cristiano Malossi[1] | Alexandru Meterez[1] | Florian Scheidegger

Correspondence

Dr. Mattia Rigotti
Email: mrg@zurich.ibm.com

[1] IBM Research AI, Zurich, Switzerland

Funding information
This contribution is part of the IM-SAFE project funded by the European Union's Horizon 2020 research and innovation programme

Abstract

The latest successes in AI have been largely driven by a paradigm known as Foundation Models (FMs), large Neural Networks pretrained on massive datasets that thereby acquire impressive transfer learning capabilities to adapt to new tasks. The emerging properties of FMs have unlocked novel tantalizing applications for instance enabling the generation of fluent text and realistic images from text descriptions. The impact of FMs on technical domains like civil engineering is however still in its infancy, owing to a gap between research development and application use cases. This paper aims to help bridge this gap and promote adoption among technical practitioners, specifically in visual inspection applications for civil engineering. For that we analyze the requirements in terms of data availability making particular use cases amenable to the pretraining/fine-tuning paradigm of FMs, i.e. situations where labeled data is scarce or costly, but unlabeled data is abundant. We then illustrate proof-of-concepts workflows using FMs, in visual inspection applications. We hope that our contribution will mark the start of conversations between AI researchers and civil engineers on the potential of FMs to accelerate workflows supporting vision tasks for maintenance inspections and decisions.

Keywords

Machine Learning, Foundation Models, Computer Vision, Visual Inspection

1 Introduction

The last decade has marked incredible technical progress in AI and Machine Learning (ML) with remarkable success stories in the fields of computer vision, speech recognition, natural language processing, and other domains. This wave of success and excitement in AI has been in large part driven by Deep Learning [1], and in particular by *Foundation Models*, deep neural networks that are trained on large broad datasets and can be deployed on a wide range of downstream tasks. This flexibility in being applied to a whole array of applications is what makes Foundation Models functionally interesting and practically valuable, and has unlocked novel tantalizing applications for instance in language and synthetic image creation by enabling the generation of fluent text [2], and realistic images from text [3]–[5].

While ML has seen increased adoption in the civil engineering domain in recent years for instance in applications like condition assessment [6], novelty detection for structural health monitoring [7] or other structural engineering tasks [8], the impact of Foundation Models specifically on civil engineering is however still in its infancy, owing to a gap between research development and application use cases.

The goal of this paper is to contribute to bridging the gap between the development of Foundation Models in AI and their practical adoption among technical practitioners, specifically in visual inspection applications for civil engineering. We do that by first analyzing the requirements in terms of data availability that renders a particular use case amenable to the use of the Foundation Model pre-training/fine-tuning paradigm, e.g. situations where data labeling is scarce or costly, but where unlabeled datasets are readily available. We then illustrate proof-of-concepts workflows for visual inspection using Foundation Models, focusing our considerations on leveraging key emerging properties of Foundation Models in an interactive human-in-the-loop visual inspection setting such as defect detection and classification for the maintenance of civil engineering structures.

2 Data requirements

Foundation Models are enabled by *transfer learning* [9], i.e. the transfer of knowledge from a source task to downstream tasks of interest, and *pretraining at scale*, i.e. pretraining the model on vast amounts of data of broad source tasks so as to supercharge the transfer learning capabilities. This is typically achieved thanks to *self-supervised learning*, a training paradigm that dispenses of potentially time-consuming *human annotations* and therefore enables scaling up the training of neural network models on large amounts of data. The availability of large amounts of data is therefore a key requirement for training a Foundation Models.

After a Foundation Model has been pretrained on large-scale unlabeled data, it can be *fine-tuned* on a downstream task of interest. This requires only a small amount of high-quality annotated data to achieve accuracies at the level of machine learning models that don't undergo large-scale self-supervised pretraining but are trained on much larger amounts of high-quality annotated data.

In this sense, Foundation Models training is an approach that trades off the need for task-specific data with the need for large amounts of data at pretraining. This is leveraged in *hierarchical self-supervised pretraining* which consists of a sequence of self-supervised training steps on decreasing amounts of increasingly task-relevant data, so as to tune the trade-off between data quantity and quality in ways that best match the data availability [10], [11].

The first conclusion in regards to data requirement is then that Foundation Models require less annotated data (used at fine-tuning) than traditional supervised learning models but need a potentially large amount of (cheaper) unlabeled data (used at pretraining). Seen from a different perspective, Foundation Models are a potentially attractive means to take advantage of large troves of available datasets that are still unlabeled and would otherwise be costly to use in a traditional supervised learning setting, because of the effort needed to annotate them.

Pretraining at scale introduces another requirement, namely the need for computer hardware and training infrastructure that supports it (see e.g. [12]). Importantly, the resulting training compute costs of Foundation Models are amortized over all the downstream tasks that can be addressed with the same model. For instance, the compute cost of pretraining GPT-3, a 175B-parameter Large Language Model by OpenAI [13], has been estimated to be around a staggering $4,600,000 [14]. However, the resulting pretrained model and its variants are now powering a wide range of downstream applications and services, including code assistant tools and chatbots, and access the model is being offered as a service through API calls.

In the next sections, we make the case for the use of Foundation Models in civil engineering. We will do that by sketching proof-of-concepts workflows and integrating them into visual inspection use cases. Our hope is that these will trigger conversations in the civil and structural engineering communities on the potential of Foundation Models to support visual inspections, and result in more consolidated workflows that might contribute to supporting maintenance inspections workloads.

3 Proof-of-Concept Workflows

3.1 Self-supervised learning on unannotated images

Computer vision, specifically in the form of deep learning models, is being increasingly deployed for aiding with the task of visual inspection of civil engineering use cases. In particular, these models can help quickly sort through large amounts of images looking for objects of interest which have a high associated risk and need to be attended to immediately, such as structural cracks and visible corroded rebar in bridges.

Nevertheless, having accurate computer vision models depends on being able to train these models on a large volume of labeled data. However, high-quality labels are difficult and costly to obtain in real world scenarios, such as in the case of visual inspection. This is because 1) objects of interest tend to be defects which are undesirable, and therefore in most use cases rarely occur. And 2) visual inspection usually relies on tasks such as object detection and instance segmentation to achieve good localization, two computer vision tasks for which labeling is notoriously time consuming and expensive. On the upside, visual inspection settings tend to yield large amounts of data due to the recent automation of data acquisition such as when using drones to scan bridges. Foundation models can learn and leverage these unlabeled data by using self-supervised learning. These in turn can be fine-tuned with few labels only all while achieving accurate performance.

In self-supervised learning (SSL) training labels are automatically generated from the data by defining a pretext task. Hence this training paradigm can leverage large amounts of data which are not labeled.

SSL has been present in the field of computer vision for many years now, with approaches such as de-noising autoencoders [15] and even earlier with the use of Siamese networks dating back to 1992 [16]. More recently we have seen a resurgence of such self-supervised pretraining approaches in computer vision PIRL [17], BYOL [18], MOCO [19] and SimCLR [20], even more recently with using vision transformer architectures ViT [21], such as in DINO [22], MSN [23], MAE [24], and SimMIM [25]. Furthermore, SSL pretraining has shown to improve overall model performance [26] and that pretrained models are very good few shot learners on downstream tasks [23].

Foundation Models for bridge defect detection

Here we report experiments with SSL pretraining for creating foundation models using Masked Auto Encoder (MAE) [24]. In this approach a large portion of the image is masked, and the model's pretext task is to learn to reconstruct the original image from the masked one. The preliminary results are promising. For example, we show that training and using a foundation model that is pretrained on concrete bridge images can boost the overall model performance on a defect detection downstream task

as can be seen in **Figure 1**.

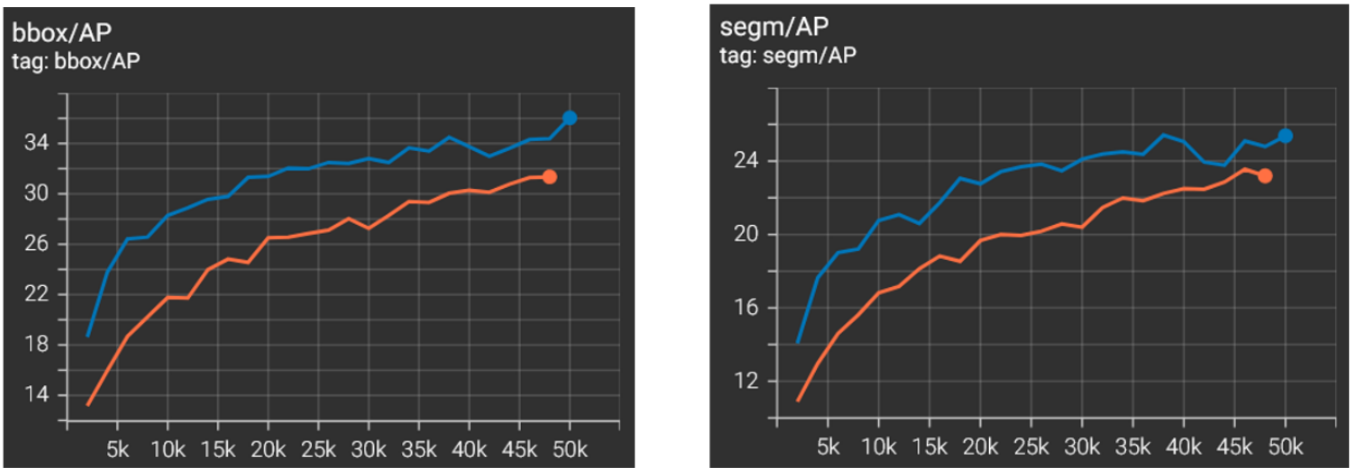

Figure 1 Overall model performance results when using a concrete bridge foundation model for fine-tuning on the task of bridge defect detection vs using a generic model for fine-tuning. On the left, the average precision of the bounding box detection, and on the right the average precision of the segmentation mask detection. The blue line represents the fine-tuned concrete bridge foundation model results, and in orange the results of a fine-tuned generic supervised pretrained model. The x-axis represents the number of training iterations.

In **Figure 1**, bbox stands for bounding box, and segm stands for segmentation mask basically the results of Object Detection and Instance Segmentation respectively. The AP score stands for Average Precision (higher is better) and which is the benchmark metric in these two tasks. The blue line represents fine-tuning the concrete bridge foundation model on the downstream task vs the orange line which represents fine tuning a generic base model for comparison.

Furthermore, we also notice that the fine-tuned concrete bridge foundation model is particularly beneficial on hard to detect defects. This can also be seen in **Figure 2**.

In **Figure 2**, we can see that using the concrete bridge foundation model does not give significant performance improvements on the algae defect which is usually a big green blob and easy to detect. Whereas it gives a significant performance boost for both the Crack and Net-Crack defects which are much finer and harder to detect.

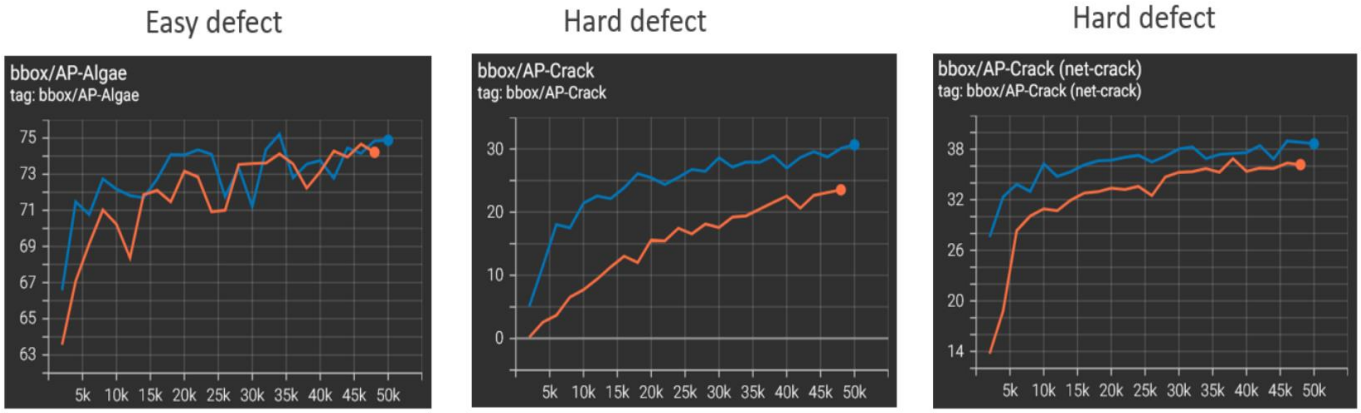

Figure 2 Per-defect model performance results when using a concrete bridge foundation model for fine-tuning on the task of bridge defect detection vs. using a generic model for fine-tuning. On the left the Algae defect results, in the center the Crack defect results, and on the right the Net-Crack defect results. The blue line represents the fine-tuned concrete bridge foundation model results, and in orange the results of a fine-tuned generic supervised pretrained model on ImageNet. The x-axis represents the number of training iterations.

3.2 Model-assisted image labeling

Model-Assisted Labeling for Visual Inspection via Explainability

As mentioned, one key advantage of Foundation Models is the decrease in the number of annotated samples needed to reach a specific accuracy in a given downstream task.

Interestingly, Foundation Models can also be useful in the complementary use case of annotating new image samples in a human-in-the-loop setting. In particular, in [27] we developed a defect annotation workflow where human annotators work to annotate segmentation masks defects on images of civil engineering structures by refining automatic annotation proposals (see **Figure 3**). Such annotation proposals are extracted using gradient-based explainability methods that produce a mask of salient pixels given an input image and a classification model trained to discriminate between different types of defects. Intuitively, the salient pixels highlighted by the explainability methods are those that are being used by the model to predict the type of defect present in the image and are therefore a good proxy for a segmentation mask of the defect.

A natural candidate for the classifier proposing the initial annotations (via explainability) is a Foundation Model

trained pretrained on large datasets and fine-tuned on discriminating between defects.

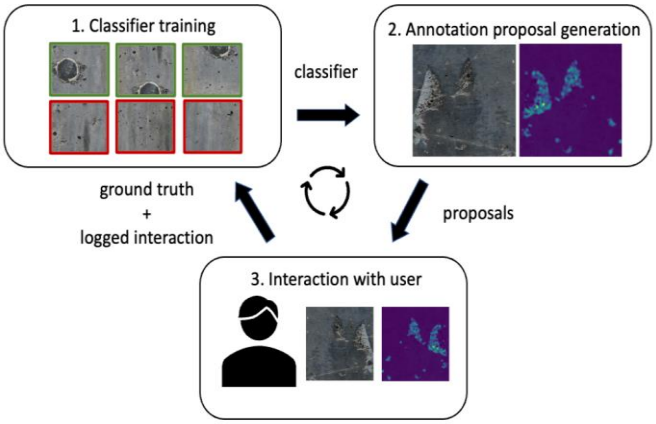

Figure 3 Overview of the model-assisted labeling framework presented in [27]. 1. A classifier is trained to discriminate between defects. 2. Gradient-based explainability applied to the trained classifier generates class-activation maps [28] . 3. The class-activation maps are processed into segmentation annotation proposals that the human annotator can refine. See [27] for more details.

Active learning for imbalanced datasets

Active Learning (AL) is another model-assisted annotation paradigm that has the scope of minimizing the number of samples that a human annotator has to label by prioritizing the labeling of samples that are predicted to be highly informative.

The goal is in other words to optimize the trade-off between the cost of additional annotations (number of annotated samples) and the accuracy of a model trained on the annotated data (which is assumed to monotonically increase as more samples are labeled). AL has been successfully applied in medical imaging [29], [30], astronomy [31], and surface defect detection [32].

In [33], we presented a method that effectively and efficiently selects minority samples from a pool of unlabeled data for large datasets suffering from heavy class imbalance. While most academic datasets are class balanced, with each class having the same number of samples, we elected to focus on the imbalanced case because it is more representative of real-world industry datasets that usually suffer from a long-tailed class distribution [34], [35]. Moreover, in these cases, the minority classes are often the most important ones. This is usually the case for civil structures: dangerous/critical defects rarely appear on drone scans of the structure as they are well maintained.

Contrary to classical AL methods that try to find informative samples from all classes, our method works by selecting samples for the minority class, investing the total labeling budget for samples that improve model performance for only that minority class. To this end, our method replaces the AL acquisition function with a binary discriminator explicitly trained in a one-vs-all fashion (minority vs. majority classes) to distinguish between unlabeled minority and majority samples.

Applying our method to our proprietary civil infrastructure dataset (see Figure 4), we show a minority class recall improvement of 32% and an overall accuracy gain of 14% compared to the best-performing traditional AL method (BALD [36]).

Combining Foundation Models and Active Learning could have a compounding impact on the number of annotated samples needed to reach a particular performance on a given downstream task: using a pretrained Foundation Model as a starting point for training the model on the initial pool of labeled data will lead to a more accurate model and, therefore, to a better selection of samples in the first annotation round. In turn, this process will lead to a more accurate model in the second Active Learning cycle. The pattern repeats until the end of the Active Learning process leading to a considerably higher sample efficiency.

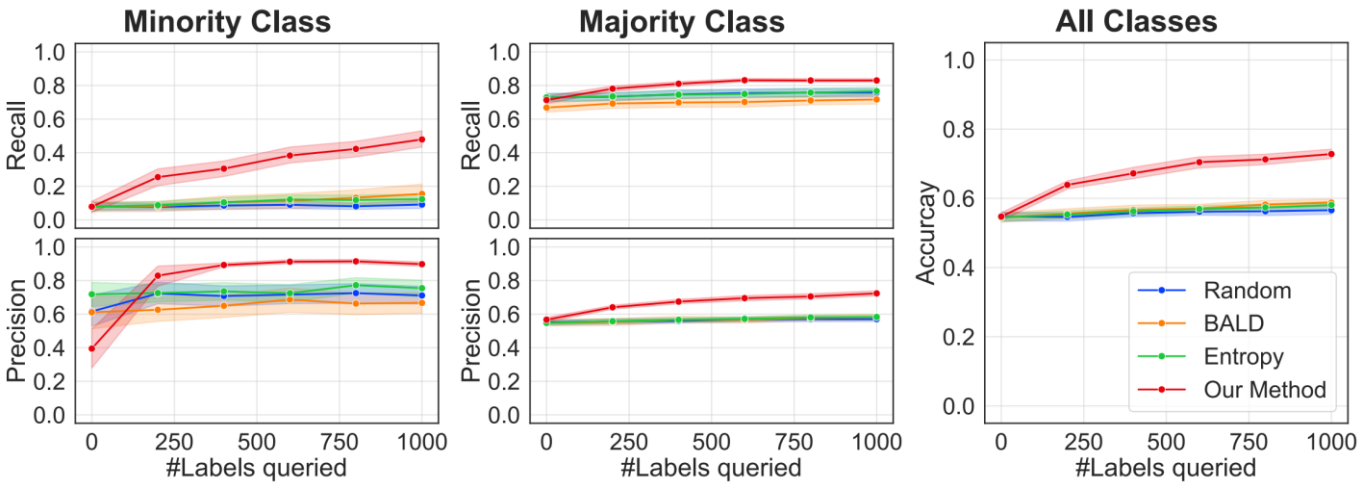

Figure 4 Results of our Active Learning for imbalance datasets on a civil engineering dataset (see [33]): absolute model performance throughout the AL process: for each cycle, after labeling 200 additional samples, we report precision and recall for the minority class, macro average precision and recall for the majority classes, and overall accuracy for our proprietary civil infrastructure dataset. Error bands show standard error of the mean (SEM).

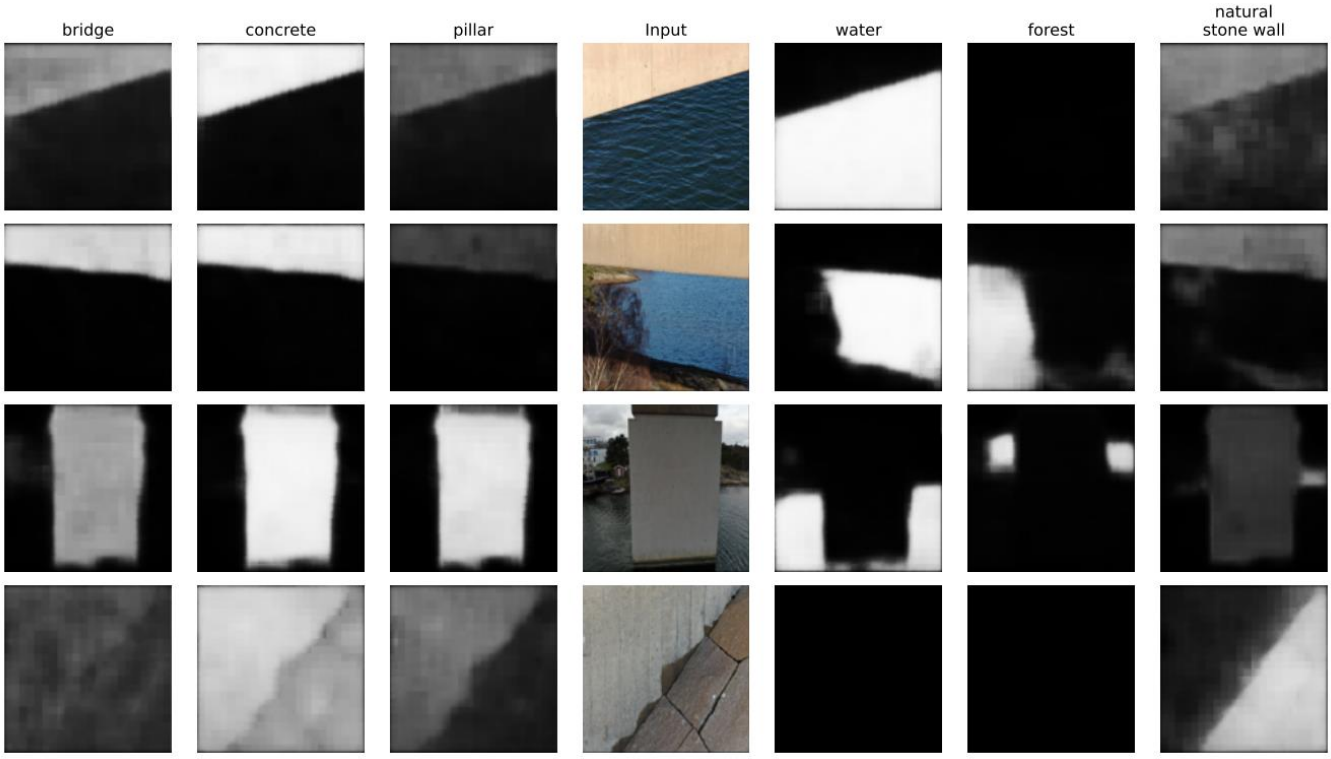

Figure 5 Predictions of ClipSeg for foreground queries (left), input (center), and background queries (right). The foreground queries are highly correlated. The background queries are selected to complement each other.

3.3 Pre-trained multi-modal segmentation models

The application of computer vision methods on images of civil engineering assets usually considerably benefits from preprocessing stages. In the case of visual inspection of civil assets such as bridges it is for instance often beneficial to separate background from foreground such that the computer vision algorithms are only applied to the planar surfaces of the bridge identified as foreground, whereas partially visible surroundings can be removed as part of the background.

ClipSeg responses to text queries

Recently, researchers have proposed a method to seamlessly perform such image segmentation like foreground/background separation by leveraging a pretrained multimodal Foundation Model known as CLIP. CLIP is a joint text-visual embedding model [37], which in practice produces a representation of images and natural language in a common vector space such that they can be used to reference visual concepts through natural language. ClipSeg [38] builds upon CLIP to obtain a system that generates image segmentations based on arbitrary natural language prompts.

Using ClipSeg out-of-the-box with queries like "bridge", "concrete", and "pillar" on typical civil engineering images already produces remarkable results in terms of separating the foreground (the concrete structure) from the background, presumably owing to the large volume of data used to train CLIP. Specifically, Figure 5 shows the predicted maps for foreground queries (left part) and background queries (right part). However, some queries, such as "concrete", produce sharper results than other queries like "pillar". Likewise, instead of only using queries that refer to foreground, combining multiple queries referring to the background seems to as well. For example, general queries such as "water" or "forest" produce a strong response on regions where they occur. Moreover, selective queries such as "natural stone wall" produce clean and problem-specific responses.

Multiple queries for improving segmentation

Even though single queries produce satisfying results, there is no single query that accurately fits all foreground/background segmentation instances. To achieve that, we suggest ensembling the response attention map of multiple queries.

Figure 6 shows the resulting foreground/background segmentation mask (left part) and the uncertainty map (right part) for three specific ensembles. We observe that for simple images, such as the top image, all three variants deliver almost perfect results that segment the bridge from the water. The uncertainty map is low, except for marking the transition region at the boundary of the segments. In contrast, the two middle images show how results improve from ensemble E_1 up to E_3. Similarly, the uncertainty is reduced (for example, between E_1 and E_3 by adding the query "forest"). Most artifacts (false-positive foreground segmentation on the left and right of the pillar of the third input image) for E_1 are marked as uncertain, henceforth indicating that the constructed uncertainty map indicates the trustworthiness of the produced output. Additionally, the last input image produces a strong response for "concrete" in all regions resulting in no background segmentation for E_1 and E_2. Introducing the problem-specific background query "natural stone wall" enabled producing a strong response to correctly segment the asset with E_3.

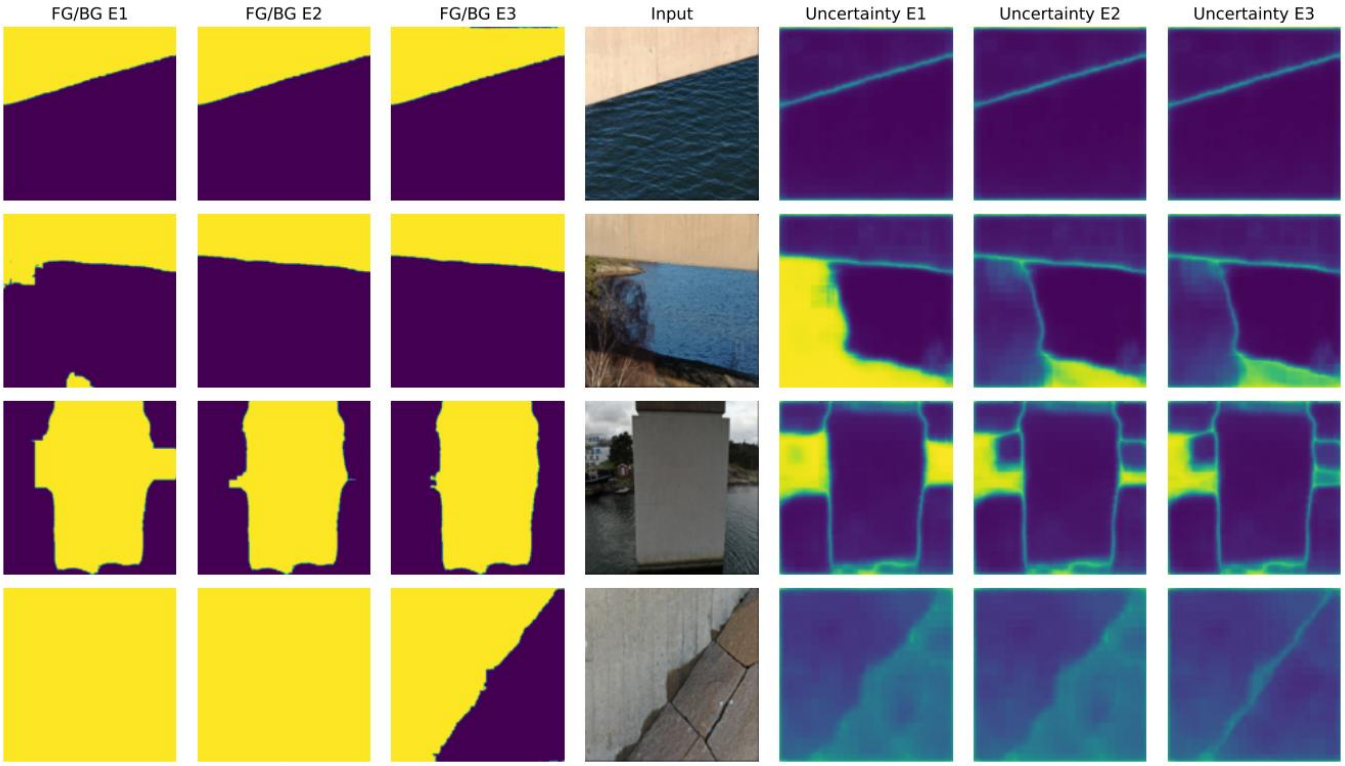

Figure 6 Ensemble results: Final foreground/background (left), input (center), and uncertainty map (right). The queries corresponding to the different ensemble settings are as follows: E_1:=(foreground=["concrete"], background=["water", "sky"]), E_2:=(foreground=["concrete"], background=["water", "sky", "forest"]), E_3:=(foreground=["concrete"], background=["water", "sky", "forest", "gravel", "natural stone wall"]). Complexer ensembles deliver better segmentations and lower uncertainty.

3.4 Generative Foundation Models for synthetic data generation

Modern image generation models have seen a significant increase in interest from researchers starting with the publication of generative adversarial network (GAN) [39] architecture in 2014. GANs were the first architecture to enable the synthetic generation of high-quality high-resolution images.

Recently, diffusion models were proposed [40] as a new breed of generative models for images rooted in non-equilibrium thermodynamics [41] that has been steadily replacing GANs thanks to their higher quality and diversity of generated samples, and higher training stability [5], [42].

Training on synthetically generated data

An interesting use case for synthetically generated data is that of data augmentation where synthetically generated samples are used to artificially increase the size of a training dataset in low data regime [43]. Besides mitigating data paucity, generating synthetic data for model training offers additional benefits, like privacy (by dissociating the synthetically generated data from sensitive information in real data) and increased fairness (by mitigating the bias issues due to the presence of underrepresented categories in the data) [44].

Example use in civil engineering

We now showcase the use of diffusion models for data augmentation as it could be applied to train a defect detection in a data paucity regime where some some defects might be underrepresented in the training dataset.

Figure 7 shows samples of "crack" objects and "concrete wall" backgrounds (i.e. intact surfaces) coming from a real dataset as well as generated by a diffusion model. These should be compared to synthetically generated images of "cracks" and "concrete walls" obtained by fine-tuning a pretrained diffusion model seen in Figure 8.

Figure 7 Real pictures of cracks (top row) and concrete walls (bottom row) selected from a real dataset.

Figure 8 Synthetically generated pictures of cracks (top row) and concrete walls (bottom row) generated with a diffusion model.

The quality, realism and diversity of synthetically generated images is striking. Moreover, notice that the visual similarity between generated background and crack images (top and bottom rows of Figure 8) is not accidental, but comes from generating both rows of images starting

from the same initialization and only modifying the prompt used to condition the generation (from "big crack on a concrete wall" to "concrete wall").

In the future, a further refinement of this method of pairing images with defects with images without defects could be potentially combined with explainability methods relying on contrastive [45] or concept-based [46] explanations in order to potentially obtain candidate segmentation masks as done in [27]. Furthermore, diffusion models could be refined to take into account the manifold geometry of synthetic images (see e.g. [47]) using for instance the formalism developed in [48]. Finally, methods based on optimal transport or its unbalanced version [49] or taking into account learning margins [50] could be used to speed up the generation of synthetic images.

4 Conclusions

In this paper we made the case for the use of AI Foundation Models in civil engineering, and in particular for their integration in visual inspection workflows. We did that by trying narrow the gap between research development and application use cases through the illustration of prototype workflows where Foundation Models are being leveraged to support automated visual inspection. We hope that this contribution will ignite durable and fruitful conversations between AI researchers and civil engineers aimed at consolidating and refining the use of Foundation Models in civil engineering.

Acknowledgement

This publication has been written as a part of IM-SAFE best practices analysis and the authors acknowledge its funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 958171. The sole responsibility for the content of this publication lies with the authors. It does not necessarily reflect the opinion of the European Union. Neither the Innovation and Networks Executive Agency (INEA) nor the European Commission are responsible for any use that may be made of the information contained therein.

The authors would like to thank Sund & Bælt Holding A/S for providing the images used in Figure 7 and in particular Finn Bormlund and Svend Gjerding for their support.

References

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[2] T. B. Brown et al., "Language Models are Few-Shot Learners." arXiv, Jul. 2020. doi: 10.48550/arXiv.2005.14165.

[3] A. Ramesh et al., "Zero-Shot Text-to-Image Generation." arXiv, Feb. 2021. doi: 10.48550/arXiv.2102.12092.

[4] C. Saharia et al., "Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding." arXiv, May 2022. doi: 10.48550/arXiv.2205.11487.

[5] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-Resolution Image Synthesis with Latent Diffusion Models." arXiv, Apr. 2022. doi: 10.48550/arXiv.2112.10752.

[6] S. Bianchi and F. Biondini, "Bridge Condition Assessment Using Supervised Decision Trees," in Proceedings of the 1st Conference of the European Association on Quality Control of Bridges and Structures: EUROSTRUCT 2021 1, 2022, pp. 1108–1116.

[7] N. Manzini et al., "An Automated Machine Learning-Based Approach for Structural Novelty Detection Based on SHM," in Proceedings of the 1st Conference of the European Association on Quality Control of Bridges and Structures: EUROSTRUCT 2021 1, 2022, pp. 1180–1189.

[8] H. Salehi and R. Burgueño, "Emerging artificial intelligence methods in structural engineering," Engineering Structures, vol. 171, pp. 170–189, Sep. 2018, doi: 10.1016/j.engstruct.2018.05.084.

[9] L. Torrey and J. Shavlik, "Transfer learning," in Handbook of research on machine learning applications and trends: Algorithms, methods, and techniques, IGI global, 2010, pp. 242–264.

[10] C. J. Reed et al., "Self-Supervised Pretraining Improves Self-Supervised Pretraining," arXiv:2103.12718 [cs], Mar. 2021, Available: https://arxiv.org/abs/2103.12718

[11] A. B. Sellergren et al., "Simplified Transfer Learning for Chest Radiography Models Using Less Data," Radiology, p. 212482, Jul. 2022, doi: 10.1148/radiol.212482.

[12] A. Bartezzaghi, I. Giurgiu, C. Marchiori, M. Rigotti, R. Sebastian, and C. Malossi, "Design of a Cloud-Based Data Platform for Standardized Machine Learning Workflows with Applications to Transport Infrastructure," in 2022 IEEE 21st Mediterranean Electrotechnical Conference (MELECON), Jun. 2022, pp. 764–769. doi: 10.1109/MELECON53508.2022.9843138.

[13] T. B. Brown et al., "Language models are few-shot learners." arXiv, 2020. doi: 10.48550/ARXIV.2005.14165.

[14] C. Li, "OpenAI's GPT-3 Language Model: A Technical Overview." https://lambdalabs.com/blog/demystifying-gpt-3, Jun. 2020.

[15] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in Proceedings of the 25th international conference on machine learning, 2008, pp. 1096–1103.

[16] S. Becker and G. E. Hinton, "Self-organizing neural network that discovers surfaces in random-dot stereograms," Nature, vol. 355, no. 6356, pp. 161–

163, 1992.

[17] I. Misra and L. van der Maaten, "Self-supervised learning of pretext-invariant representations," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 6707–6717.

[18] J.-B. Grill et al., "Bootstrap your own latent-a new approach to self-supervised learning," Advances in neural information processing systems, vol. 33, pp. 21271–21284, 2020.

[19] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 9729–9738.

[20] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in International conference on machine learning, 2020, pp. 1597–1607.

[21] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv preprint arXiv:2010.11929, 2020.

[22] M. Caron et al., "Emerging properties in self-supervised vision transformers," in Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 9650–9660.

[23] M. Assran et al., "Masked siamese networks for label-efficient learning," in Computer vision–ECCV 2022: 17th european conference, tel aviv, israel, october 23–27, 2022, proceedings, part XXXI, 2022, pp. 456–473.

[24] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 16000–16009.

[25] Z. Xie et al., "Simmim: A simple framework for masked image modeling," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 9653–9663.

[26] C. J. Reed et al., "Self-supervised pretraining improves self-supervised pretraining," in Proceedings of the IEEE/CVF winter conference on applications of computer vision, 2022, pp. 2584–2594.

[27] K. Janouskova, M. Rigotti, I. Giurgiu, and C. Malossi, "Model-Assisted Labeling via Explainability for Visual Inspection of Civil Infrastructures," in Computer Vision 2022 Workshops: Tel Aviv, Israel, October 23, 2022, Proceedings, Part VII, 2023, pp. 244–257.

[28] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization," International Journal of Computer Vision, vol. 128, no. 2, pp. 336–359, Feb. 2020, doi: 10.1007/s11263-019-01228-7.

[29] X. Shi, Q. Dou, C. Xue, J. Qin, H. Chen, and P.-A. Heng, "An Active Learning Approach for Reducing Annotation Cost in Skin Lesion Analysis," in Machine Learning in Medical Imaging, 2019, pp. 628–636.

[30] W. Li et al., "PathAL: An Active Learning Framework for Histopathology Image Analysis," IEEE Transactions on Medical Imaging, vol. 41, no. 5, pp. 1176–1187, May 2022, doi: 10.1109/TMI.2021.3135002.

[31] J. W. Richards et al., "Active learning to overcome sample selection bias: Application to photometric variable start classification," The Astrophysical Journal, vol. 744, no. 2, p. 192, Dec. 2011, doi: 10.1088/0004-637X/744/2/192.

[32] C. Feng, M.-Y. Liu, C.-C. Kao, and T.-Y. Lee, "Deep Active Learning for Civil Infrastructure Defect Detection and Classification," in Computing in Civil Engineering 2017, Jun. 2017, pp. 298–306. doi: 10.1061/9780784480823.036.

[33] T. Frick, D. Antognini, M. Rigotti, I. Giurgiu, B. Grewe, and C. Malossi, "Active Learning for Imbalanced Civil Infrastructure Data," in Computer Vision 2022 Workshops: Tel Aviv, Israel, October 23, 2022, Proceedings, Part VII, 2023, pp. 283–298.

[34] Y. Zhang, B. Kang, B. Hooi, S. Yan, and J. Feng, "Deep Long-Tailed Learning: A Survey." arXiv, Oct. 2021. Accessed: Jul. 04, 2022. [Online]. Available: http://arxiv.org/abs/2110.04596

[35] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, and S. X. Yu, "Large-Scale Long-Tailed Recognition in an Open World," 2019, pp. 2537–2546.

[36] N. Houlsby, F. Huszár, Z. Ghahramani, and M. Lengyel, "Bayesian Active Learning for Classification and Preference Learning." arXiv, Dec. 2011. doi: 10.48550/arXiv.1112.5745.

[37] A. Radford et al., "Learning Transferable Visual Models From Natural Language Supervision," in Proceedings of the 38th International Conference on Machine Learning, Jul. 2021, pp. 8748–8763.

[38] T. Lüddecke and A. Ecker, "Image segmentation using text and image prompts," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), 2022, pp. 7086–7096.

[39] I. J. Goodfellow et al., "Generative adversarial networks." arXiv, 2014. doi: 10.48550/ARXIV.1406.2661.

[40] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," CoRR, vol. abs/2006.11239, 2020, Available: https://arxiv.org/abs/2006.11239

[41] J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep Unsupervised Learning using Nonequilibrium Thermodynamics," arXiv:1503.03585 [cs, stat], Mar. 2015, Available: https://arxiv.org/abs/1503.03585

[42] P. Dhariwal and A. Nichol, "Diffusion models beat GANs on image synthesis." arXiv, 2021. doi: 10.48550/ARXIV.2105.05233.

[43] C. Chadebec, E. Thibeau-Sutre, N. Burgos, and S. Allassonnière, "Data Augmentation in High Dimensional Low Sample Size Setting Using a Geometry-Based Variational Autoencoder." arXiv, Jun. 2022. doi: 10.48550/arXiv.2105.00026.

[44] I. Padhi et al., "Tabular Transformers for Modeling Multivariate Time Series," in ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Jun. 2021, pp. 3565–3569. doi: 10.1109/ICASSP39728.2021.9414142.

[45] A. Dhurandhar et al., "Explanations based on the Missing: Towards Contrastive Explanations with Pertinent Negatives," arXiv:1802.07623 [cs], Oct. 2018, Available: https://arxiv.org/abs/1802.07623

[46] M. Rigotti, C. Miksovic, I. Giurgiu, T. Gschwind, and P. Scotton, "Attention-based Interpretability with Concept Transformers," in International Conference on Learning Representations (ICLR), 2022.

[47] C. Chadebec and S. Allassonnière, "Data Augmentation with Variational Autoencoders and Manifold Sampling," in Deep Generative Models, and Data Augmentation, Labelling, and Imperfections, Springer, 2021, pp. 184–192.

[48] M. Rigotti and F. Debbasch, "An H-theorem for the general relativistic Ornstein-Uhlenbeck process," Journal of Mathematical Physics, vol. 46, p. 103303, 2005, doi: 10.1063/1.2038627.

[49] Y. Mroueh and M. Rigotti, "Unbalanced Sobolev Descent," in Advances in Neural Information Processing Systems (NeurIPS), 2020, vol. 34.

[50] O. Barak and M. Rigotti, "A Simple Derivation of a Bound on the Perceptron Margin Using Singular Value Decomposition," Neural Computation, vol. 23, no. 8, pp. 1935–1943, Sep. 2011, doi: 10.1162/NECO_a_00152.