OsiriXgrpc: Rapid development and deployment of state-of-the-art artificial intelligence for clinical practice.

Timothy Sum Hon Mun[1], Richard Holbrey[1], Sue Chua[2], Paul Huang[3], Christina Messiou[1,4], Simon J Doran[1], Matthew D Blackledge[1,*]

[1]Division of Radiotherapy and Imaging, The Institute of Cancer Research, London, SW7 3RP.
[2] Department of Nuclear Medicine and PET/CT, The Royal Marsden Hospital, Sutton, SM2 5PT.
[3] Division of Molecular Pathology, The Institute of Cancer Research , London, SW7 3RP
[4] MRI Unit, The Royal Marsden Hospital, Sutton, SM2 5PT
*Correspondence: matthew.blackledge@icr.ac.uk

Abstract

We describe and evaluate our open-source project, **OsiriXgrpc**, for rapid development and prototyping of artificial intelligence (AI) tools in radiology. Our platform enables inter-process communication between a highly popular radiological viewing platform, OsiriX, on the network using the Google Remote Procedure Call (gRPC) infrastructure. Importantly, this provides OsiriX with access to popular AI libraries including TensorFlow, Torch, and MONAI via Python deployed on systems that contain dedicated graphical processing units (GPUs) for inference of complex deep-learning (DL) models. Currently, OsiriXgrpc only supports Python but it can be easily extended to other programming languages thanks to the flexibility of gRPC. We provide benchmark tests on the transfer rate of images between an OsiriX server and Python client, comparing three different connection methods: (i) a local Python process on the same machine as the OsiriX server, (ii) on a remote GPU-enabled Linux kernel directly connected through a local Ethernet port, and (iii) on a remote Linux kernel connected using a domestic WiFi network. Our initial results demonstrate the rapid communication achievable using gRPC, with average transfer speeds of 627 and 152 images/second over wired and wireless networks respectively. We then demonstrate the utility of OsiriXgrpc for deploying AI models through two use cases: (i) evaluating two deep learning (DL) models for automatic spleen and multi-organ segmentation on CT images using UNet and UNEt TRansformer (UNETR) architectures, and (ii) using a Siamese Network architecture for whole-body CT-to-MRI slice matching.

Introduction

Imaging is essential within patient healthcare and medical research. Due to a surge in the use of positron emission tomography (PET), X-ray computerised tomography (CT), and magnetic resonance imaging (MRI) over the last two decades, there is an urgent need for tools that can quickly and accurately help radiologists assess large and very complex imaging datasets (Pichler, Judenhofer, and Pfannenberg 2008). Artificial intelligence (AI) is demonstrating ever-increasing importance within medical imaging due to its ability to automate sometimes complex and repetitive tasks that would otherwise need intervention by an expert radiologist. By improving the workflow efficiencies within radiology departments there is hope that AI can reduce the global cost of healthcare and help alleviate the workflow of busy practitioners whose time could be better spent providing specialist care to those patients most at need and improving overall patient outcome (Tang et al. 2018; European Society of Radiology (ESR) 2019). Due to improvements in computer hardware and algorithm design, deep-learning has demonstrated state-of-the-art performance for many radiological tasks including automatic region-of-interest (ROI) delineation (often referred to as 'segmentation'), tissue classification, and spatial registration of different imaging modalities (Jiang et al. 2017; Panch, Mattie, and Celi 2019; Subbaswamy and Saria 2020; Yu, Beam, and Kohane 2018). Moreover, with the advent of techniques including radiomics (Gillies, Kinahan, and Hricak 2016; Kumar et al. 2012), there is hope that these tools can uncover complex signatures not visible to the human eye that can be used for tasks such as predicting patient outcome and/or response to novel treatments.

Unfortunately, the majority of AI tools will never reach clinical practice despite the great promise they may show during research trials. This is in part due to the inherent difficulty in meeting the requirements of medical regulatory bodies, who stipulate extensive clinical testing and development of software to specific standards (Dikici et al. 2020; Kotter and Ranschaert 2021). Therefore, robust software that allows rapid prototyping, training, deployment and clinical testing of AI models, paired with an interface familiar to clinicians will be highly valuable for this field.

OsiriX is a popular FDA-approved and CE-certified medical image viewing platform that is built using the Mac-native Objective-C programming language and libraries. It has been used by clinicians worldwide and can perform many of vital medical image viewing tasks including 2D/3D/4D data visualisation, DICOM (Bidgood Jr. et al. 1997; Gibaud 2008) database management, and provide a query/retrieve node for Picture Archiving and Communication Systems (PACS). Importantly, the platform enables development of bespoke plugins that can assist in performing specific tasks. However, these plugins must be written in Objective-C, which does not provide strong support for deployment of

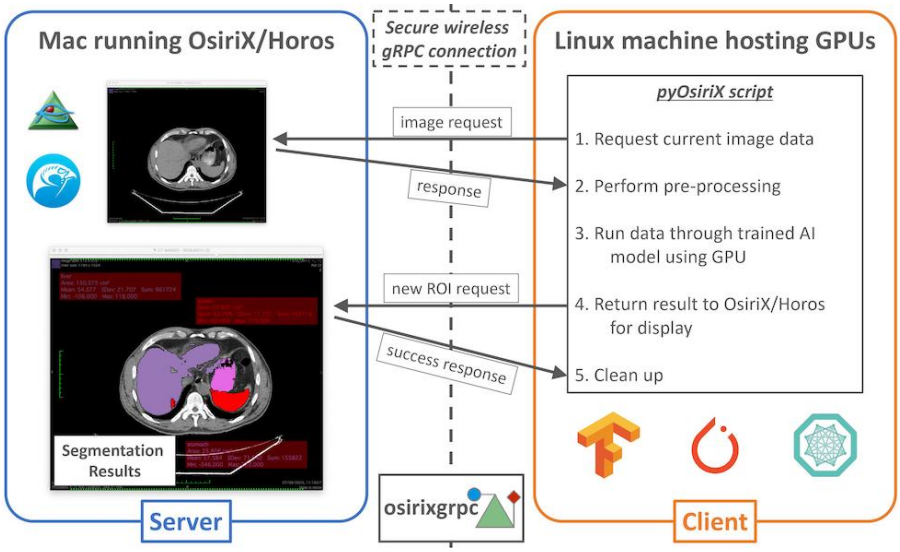

Figure 1: An illustration of OsiriXgrpc server communicating with a Linux kernel run on a machine connected via a secure wireless link (WiFi) using gRPC messaging. Importantly, the client is able to use any language and available libraries for processing and pass the results back to OsiriX for display. Importantly, these libraries include TensorFlow, Torch and MONAI.

AI algorithms, nor is it particularly easy to develop.

We originally invented pyOsiriX (Blackledge et al. 2016) to solve the issues of long and complex plugin design by allowing users to rapidly develop plugins using Python rather than Objective-C. Unfortunately, this tool suffered from a number of fundamental drawbacks that limits it's current use: (i) only supports Python 2.7, (ii) runs plugins on the main thread leading to freezing of the main software when performing long calculations, and (iii) did not provide an adequate development environment.

OsiriXgrpc

In this paper, we present OsiriXgrpc as a major update to our previous solution; we use the Google remote procedure call (gRPC) architecture to facilitate inter-process communication between OsiriX and another Python process, which can be hosted either on the same machine or on a GPU-enabled computer/cluster of machines (with potentially different operating systems) connected via a relevant network. Importantly, the technology enables users to use *any* version of Python they choose, along with any relevant 3rd-party research libraries such as Tensorflow (Abadi et al. 2015) and Torch (Paszke et al. 2019). We anticipate that OsiriXgrpc will support AI researchers efficiently deploy medical imaging deep-learning models within a radiological workflow, enabling effective collaboration with expert clinicians and feedback on model accuracy for iterative model refinement. This project will be open source and freely available to the imaging community for research use.

gRPC is an open source, high performance HTTPS remote procedure call framework developed by Google that can run on any enabled environment (https://grpc.io/). It uses language-independent protocol buffers to serialize the message interchange process, allowing server-client communication between different systems and software languages. Figure 1 presents an illustration of the communication pathway between a Linux client running a set of Python commands, sending requests to the OsiriX server, and waiting for a response before continuing with operation. Through this mechanism is possible to query data from OsiriX, process those data in any desired way, and subsequently send a request back to OsiriX to view the result (in this case as automatically generated regions of interest).

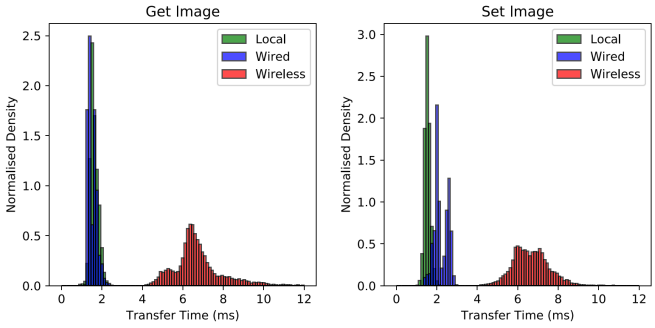

Figure 2: Distributions of image transfer time (in milliseconds) between client and the OsiriXgrpc for three different experimental setups. Note that running the gRPC service over a wired network does not reduce performance considerably, whilst the time taken to send data over our private wireless network was considerably longer.

Experiments

We performed several experiments to demonstrate the capabilities of OsiriXgrpc for enhancing the workflow of clinicians via embedded AI within OsiriX.

Benchmark Message Transfer tests

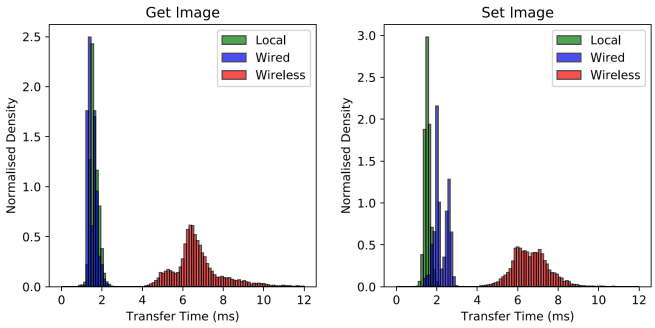

Figure 3: Distributions of image transfer time (in milliseconds) between client and the OsiriXgrpc for three different experimental setups. Note that running the gRPC service over a wired network does not reduce performance considerably, whilst the time taken to send data over our private wireless network was considerably longer.

We tested the gRPC message passing speeds between server and client using three difference connection types: (i) **local** hosted client (no access to GPU architecture), (ii) over a **wired** ethernet connected, peer-to-peer network operating at 1000 Mb/s (client housed a NVIDIA RTX6000 GPU card), and (iii) over a domestic **wireless** WiFi network (2.4GHz channel). In all cases, an insecure network was used to test peak performance (experiments performed behind a local firewall). The time taken by the client to (a) get the currently displayed image within OsiriX (always size 134×108), or (b) set the same image to the same pixel values was recorded (image parsing is currently the most data-heavy operation available to OsiriXgrpc). The same operation was performed 10^5 times for each experiment to acquire statistics on image transfer times.

Results Figure 3 demonstrates histograms of transfer times (in ms) for all three use cases, for both receiving and sending image data. The average transfer rates of image data retrieval ($\pm$ standard deviation) were 626.7 ± 77.3, 670.1 ± 86.4, and 151.5 ± 27.5 images/second for experiments (i)-(iii) respectively, whilst for image data sending they were 656.8 ± 63.4, 462.8 ± 78.1, and 151.6 ± 23.4 images/second. The transfer time between local and wired connections for gRPC message passing shows very little network overhead and either technique will likely be the method of choice when using OsiriXgrpc within a clinical research setting.

Healthy Organ Segmentation with UNETR and UNET in MONAI

To provide an example of the convenient interaction between OsiriX and a GPU client providing model inference, we integrated OsiriX with the MONAI medical deep learning library using OsiriXgrpc to perform automatic spleen segmentation from abdominal CT scans. We evaluated two deep-learning architectures: (i) a standard UNet model (Ronneberger, Fischer, and Brox 2015) and, (ii) a variation of the traditional UNet, UNETR, that utilizes a Visual Transformer (Dosovitskiy et al. 2021) as the backbone image encoder to learn sequence representations of the input volume (Hatamizadeh et al. 2021). Both models were previously trained using the MONAI deep-learning platform (Consortium 2020); the UNet model was trained on the spleen segmentation dataset from the Medical Segmentation Decathlon (Antonelli et al. 2021), whilst the UNETR model was trained on the multi-organ delineation dataset from the "Multi-Atlas Labeling Beyond the Cranial Vault" challenge (Segmentation 2015). For the latter method, only the spleen segmentation was used during inference. Both methods were tested on the same, previously unseen CT dataset of a patient with soft-tissue sarcoma (Vallières et al. 2015) from The Cancer Imaging Archive (Clark et al. 2013) (stored within OsiriX). These data had the following image acquisition parameters: isotropic in-plane spatial resolution: 0.98 mm, slice thickness 3.75 mm, resampled to a resolution of $1.5 \times 1.5 \times 2.0$ mm^3. Inference was performed on a PC running Ubuntu 18.04, with a NVIDIA RTX6000 GPU card with 24GB of memory; OsiriXgrpc was run using the wireless network connection for convenience.

Results Figure 4 shows the resulting spleen segmentations for both models within OsiriX. Total data transfer time to GPU client from OsiriX was 24 seconds in both cases, whilst inference times on the GPU were 4.4 and 8.3 seconds for the UNet and UNETR models respectively. Using OsiriXgrpc, integration of these models into OsiriX was trivial. The resultant ROIs within OsiriX could be further refined and corrected by radiologists to (i) improve accuracy of delineation, (ii) negate the need for radiologists to delineate organs from scratch, and (iii) potentially improve future model accuracy through retraining.

Slice Synchronization using Siamese Networks

An increasingly common practice in clinical radiology is to use multi-modal imaging to assess patient status. For example, whole-body MRI can often be combined with other imaging modalities such as PET/CT for evaluation of oncological diseases including Lymphoma (van Ufford et al. 2011). An important aspect of assessing multi-modal images is slice matching, that is, manually identifying the same patient position on the available image datasets so that the radiologist can subsequently scroll through axial imaging slices synchronously. Slice-matching in this way enables the radiologist to combine the strengths and weaknesses of imaging modalities to improve sensitivity and specificity of disease assessment. In addition to this, patients may undergo imaging at multiple time-points to measure response to treatment (Eisenhauer et al. 2009), or as part of the surveillance strategy for their disease (Kang et al. 2021). An automated approach to slice matching is therefore highly welcomed to accelerate the radiology workflow. For this purpose, we investigated the use of Similarity Learning (Cheng, Zhang, and Zheng 2018) by training a Siamese Network model with an architecture similar to the one described in the original paper by Koch *et. al.* (Koch et al. 2015).

We developed a generalised model for both CT to CT

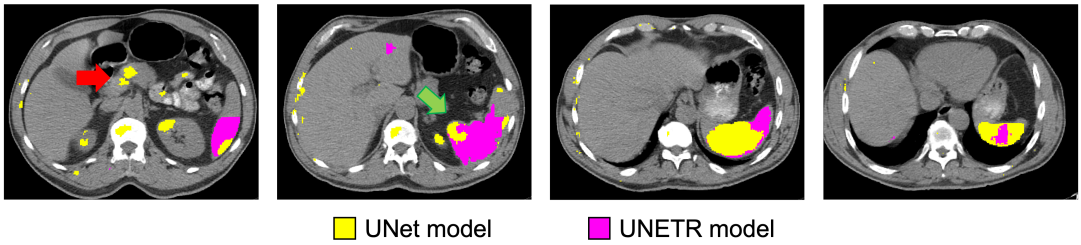

Figure 4: Example segmentation results from both trained architectures on previously unseen data of a patient with soft-tissue sarcoma. The UNet model tends to over-segment regions outside the spleen (red arrow) and under-segment the spleen (green arrow) compared to the UNETR model. Visualisation of resulting regions through OsiriX or Horos enables editing of segmentation results.

(longitudinal assessment) and CT to MRI (multi-modal assessment) axial slice matching, using an existing institutional dataset of 31 patients with lymphoma who underwent whole-body PET/CT and T1-weighted MRI for assessment of disease before and after chemotherapy. We excluded patients for whom the cerebellum was not clearly present on either CT or MRI images, had missing CT or MRI scans (due to patient withdrawal from the study), where a large mismatch of the angle/tilt between CT and MR images was observed, and any patients with metal implants. This resulted in 23/31 patient datasets for use in our study. T1-weighted MRI was conducted on a 1.5T Magnetom Aera (Siemens Healthcare, Germany) using the following protocol: spoiled gradient echo sequence with repetition time (TR) 386 ms, anterior-posterior phase-encoding direction, echo time 4.8 ms, 70 flip-angle, 256×256 acquisition matrix size interpolated to 512×512, interpolated resolution 0.74×0.74 - 0.82×0.82 mm^2, slice thickness 5mm, and parallel imaging acceleration factor R=2 (GRAPPA reconstruction). PET/CT imaging was performed on a Gemini scanner (Philips, United States) using the following acquisition parameters (CT only): acquisition type Helical, slice thickness 3 - 6.5 mm, matrix size 512×512, in-plane pixel spacing 0.74×0.74 - 1.17×1.17 mm^2, exposure 26-80 mAs.

The original study was reviewed and approved by the Committee for Clinical Research at the Royal Marsden Hospital. The patients/participants provided written informed consent to participate in this study.

Data Preprocessing

To generate ground truth data, we manually annotated three landmarks in the craniocaudal direction (head to foot) on sagittal reconstructions of both CT and MRI studies: (i) tip of the peg of the C2 vertebral body, (ii) top of the sternum and (iii) center of the L5 vertebral body. To normalize the slice locations across patients (to allow both inter- and intra-patient slice matching), we used a linear transformation to convert the slice location of the C2 vertebral body tip and top of the sternum to 0 and 1 respectively, and a second linear transformation such that the center of the L5 vertebral body had a slice location of 2 (this methodology is illustrated in Figure 6). Using these transformations, slice matching became a matter of equating the normalized slice location across modalities, time points, or patients. We generated 3896 training examples using this approach. The positive pairs consists of 1448 CT-CT and 1448 CT-MRI that are uniformly sampled across all possible slice locations. The negative pairs consists of 500 CT-MRI and 500 CT-CT that are randomly sampled between different slice locations from the patients. Subsequently, these datasets were resampled to 105×105 pixels. No data augmentation was performed. To ensure consistency of generated data, we visually verified all slice matching results generated using this technique.

Model Design, Training and Deployment

An illustration of the model architecture is provided in Figure 5. The Siamese Network model that was used for the results in this paper was trained with the following parameters: Contrastive loss, Learning Rate = 0.0005, 5000 Epochs, Adam optimizer. Due to the variability of the number of slices between different patients and different modalities, we considered only slices that lay within the three landmarks during training. This ensured that the matching slices that are generated were not outside of the range of the chosen landmarks.

To synchronise two whole-body imaging datasets during deployment, we evaluate the pairwise mean-absolute distance (MAD) generated by the Siamese network between each slice in one image series (moving series) and all slices within the second image series (fixed series). The slice which has the smallest MAD is considered to be the closest matching slice. Finally, to reduce influence of outlier predictions, we perform linear Huber regression of the resulting predicted matching slice positions in the moving image series against slice positions of the fixed image series; this provides a linear mapping of slice location. The trained slice matching model was deployed into OsiriX using OsiriXgrpc for testing in the radiologist's workflow.

Results Figure 7 shows the MAD between the different slice locations for a T1-weighted MRI data for one patient (moving series) and a single fixed CT image of a second patient. It can be seen that the smallest MAD accurately predicts the matching CT slice in this example.

Figure 8 demonstrates the full estimation of MRI-CT slice matching for two exemplar cases: (i) where MRI and CT were preformed on different patients (difficult case), and (ii)

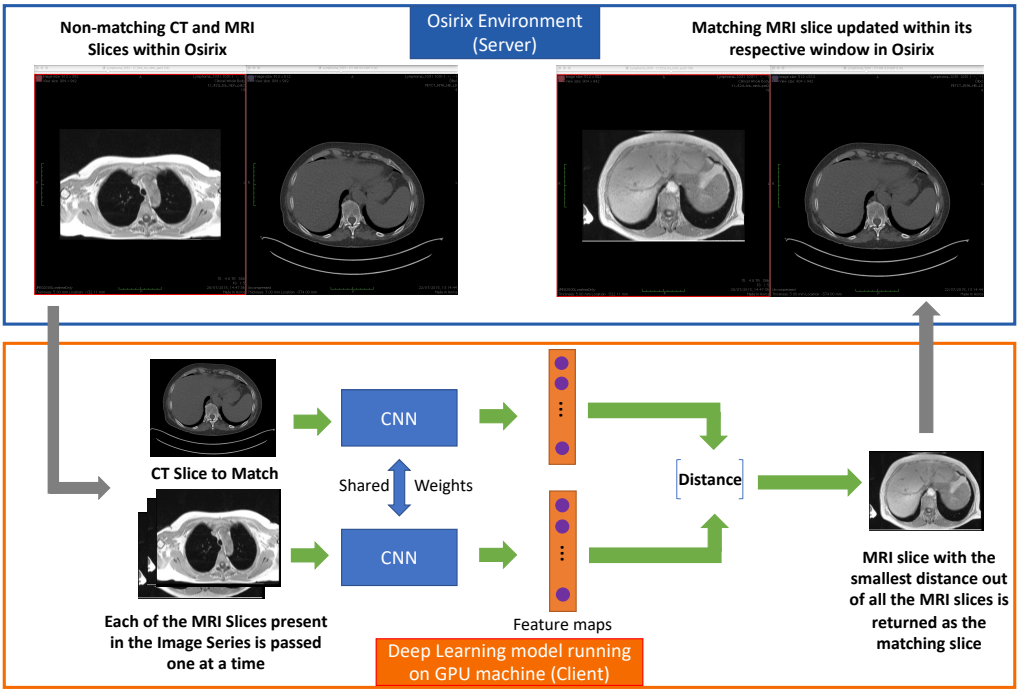

Figure 5: Process flow of automated slice synchronization within OsiriX using OsiriXgrpc.

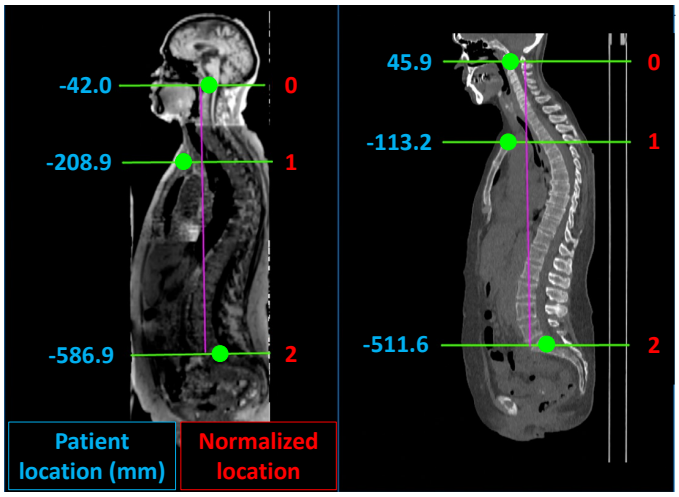

Figure 6: Example of the three annotated landmarks for both CT and MRI studies: (1) tip of the peg of the C2 vertebral body, (2) top of the sternum and (3) center of the L5 vertebral body. The values in red are the normalized slice locations and the values in blue are actual slice locations.

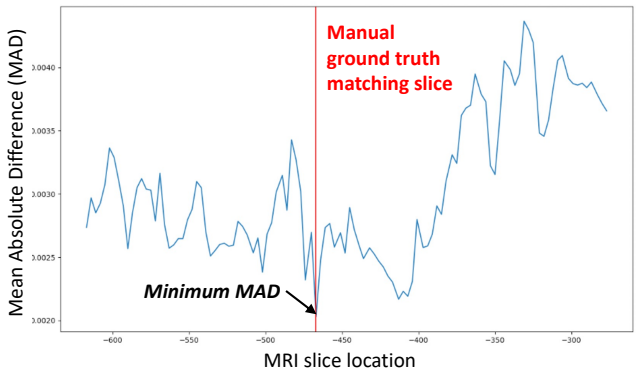

Figure 7: Mean absolute distance (MAD) between feature vectors extracted from the Siamese network for each MRI slice location in the moving image series with a single CT slice location in the fixed image series (at slice location -353mm). The MRI slice with smallest MAD (location -470mm in this case) should be the MRI slice that is most similar to that CT slice (manual ground truth represented by the red vertical line).

where MRI and CT have been acquired in the same patient (easier case). It can be seen that the fitted Huber regression model for the predicted MRI slice locations follow the manual gound truth closely in both cases. The Huber regression model for different patients generally under-predicts the ground truth slice location at all slice locations.

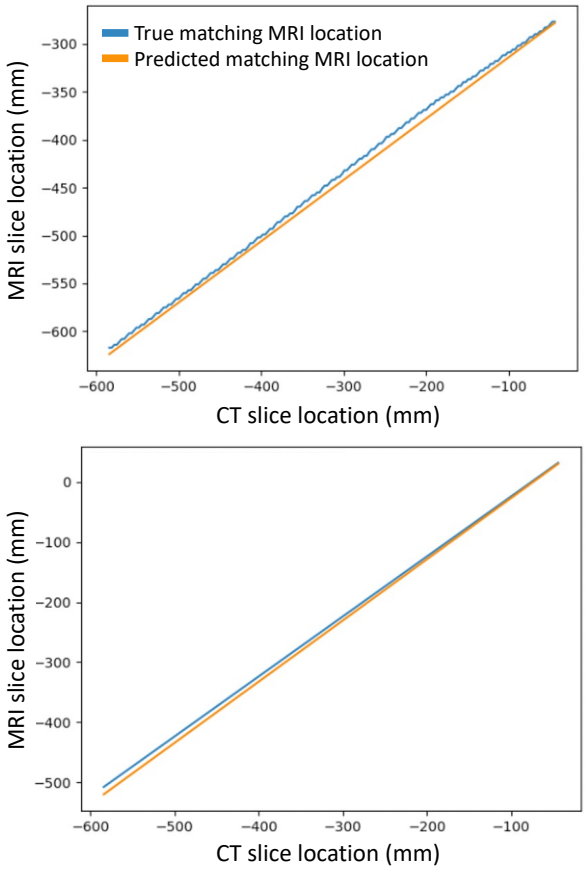

Figure 8: Comparison between the true MRI and the predicted MRI slice location for different CT slice locations when CT and MRI has been acquired in different patients (**top**) and within the same patient (**bottom**).

Discussions

A key advantage of OsirixgRPC is that with negligible costs for the developer, it is possible to implement client messages in any language supported by gRPC (as of writing this article, this includes C#, C++, Dart, Go, Java, Kotlin, Node.js, Objective-C, PHP, Python and Ruby). In addition, gRPC also enables secure transfer of messages over a network via the SSL encryption protocol. Such approaches are vital when running an OsiriXgrpc client over a network rather than as a local client on the same machine as the OsiriX server. This flexibility will allow us and the open source community to easily add new features to OsirixgRPC in the future.

We used the sending and retrieval of images for our initial benchmark tests to provide an approximation of the transfer times expected when using OsirixgRPC for the deployment of AI models within OsiriX depending on different network configurations. The low network overhead in transferring data through gRPC message passing is important since any long transfer or processing times can negatively impact the clinicians' workflow (Kansagra, Liu, and John-Paul 2016). More thorough benchmark tests could further elucidate the impact of OsiriXgrpc processing times on the radiologist workflow.

The positive segmentation results of the UNETR model will be valuable in assisting radiologists if it can be easily deployed into a platform that is familiar to them. We provided a working example of integrating it into OsiriX to provide automatic delineation of the spleen using UNETR. Additionally, the radiologist can easily correct the predicted segmentations within OsiriX to improve delineation accuracy based on their expertise and also annotate the ROIs in OsiriX. This could provide a feedback mechanism to generate further training examples and improve future model accuracy.

We have explored the use of Siamese networks to provide automatic inter- and intra-modality slice matching in whole-body medical imaging (focusing on T1-weighted MRI and CT). We demonstrated excellent results using this technique, even in the difficult case of different patients and different modalities. By combining this approach with Huber regression of the predicted slice locations within OsiriXgrpc allowed us to expose this slice synchronisation functionality within the OsiriX with ease.

There are several limitations of the current similarity learning model. Firstly, it only works with slices that lie between the tip of the peg of C2 and L5 vertebrae as this was the region that the model was trained on. The landmark definition in the training data was performed by a non-radiologist and thus may be subject to higher variability. The impact of inter- and intra-reader variability of landmark definition on the performance of the model should be further investigated.

Image artefacts that are present in the the CT and MRI images resulting from different patient positions such as patients having their arms up in the CT scans while having their arms down in the MRI scans can make it harder to slice match. Furthermore, our slice-matching model only works for imaging acquired in the axial plane.

Training a generalised slice-matching model by combining mixed pairs of CT-CT and CT-MRI either from the same of different patients increases the required complexity of the model significantly. Improvement in model accuracy may be easily gained by training individual networks for the problem at hand (e.g. within-patient CT-CT pairs, within-patient CT-MRI, between patient MRI-MRI and so on).

The aforementioned issue and the performance of the model may be improved by training a more complicated similarity learning architecture such as triplet (Hoffer and Ailon 2015) or quadruplet networks (Chen et al. 2017), or using a different backbone network for the siamese network.

Conclusions

The potential benefits of OsiriXgrpc for agile development and deployment of AI models using multiple software languages are substantial ranging from building better predictive models using domain knowledge from clinicians through an environment familiar to them, faster adaptive experimental cycles, and easier acceptance of AI tools into clinical or research workflows.

We have also provided two case studies using AI with OsiriXgrpc: (i) deployment of UNet and UNETR models for segmentation of spleen on CT studies on a GPU-enabled machine, and (ii) automatic slice matching for whole-body CT and MRI datasets using Similarity Learning. These case-studies provide a snippet of the potential for OsiriXgrpc to bridge the gap between state-of-the-art AI capcabilities and deployment on radiological workstations. As of the date of this paper, no NVIDIA graphics cards are available on Mac computers, yet OsiriXgrpc provides the capability of fast inference via network connection to GPU-enabled client machines. We have shown the network overhead is small especially when using a Ethernet peer-to-peer connection when passing the images from an OsiriX service to a client on the network. Our future roadmap includes provision of secure SSL channels for increased data security that will be important to ensure data compliance with hospital data, addition of a Python wrapper for OsiriXgrpc message passing to the Python Package Index, example scripts of AI models using OsiriXgRPC and release of the source code to the community.

Acknowledgements

This work was supported by the International Accelerator Award funded by Cancer Research UK [C56167/A29363], Associazione Italiana per la Ricerca sul Cancro [AIRC - 24297] and Fundacion Científica – Asociacion Espanola Contra el Cancer [Foundation AECC - GEACC19007MA]. We also acknowledge funding from the MedTech SuperConnector scheme.

References

Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G. S.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Goodfellow, I.; Harp, A.; Irving, G.; Isard, M.; Jia, Y.; Jozefowicz, R.; Kaiser, L.; Kudlur, M.; Levenberg, J.; Mané, D.; Monga, R.; Moore, S.; Murray, D.; Olah, C.; Schuster, M.; Shlens, J.; Steiner, B.; Sutskever, I.; Talwar, K.; Tucker, P.; Vanhoucke, V.; Vasudevan, V.; Viégas, F.; Vinyals, O.; Warden, P.; Wattenberg, M.; Wicke, M.; Yu, Y.; and Zheng, X. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Software available from tensorflow.org.

Antonelli, M.; Reinke, A.; Bakas, S.; Farahani, K.; AnnetteKopp-Schneider; Landman, B. A.; Litjens, G.; Menze, B.; Ronneberger, O.; Summers, R. M.; van Ginneken, B.; Bilello, M.; Bilic, P.; Christ, P. F.; Do, R. K. G.; Gollub, M. J.; Heckers, S. H.; Huisman, H.; Jarnagin, W. R.; McHugo, M. K.; Napel, S.; Pernicka, J. S. G.; Rhode, K.; Tobon-Gomez, C.; Vorontsov, E.; Huisman, H.; Meakin, J. A.; Ourselin, S.; Wiesenfarth, M.; Arbelaez, P.; Bae, B.; Chen, S.; Daza, L.; Feng, J.; He, B.; Isensee, F.; Ji, Y.; Jia, F.; Kim, N.; Kim, I.; Merhof, D.; Pai, A.; Park, B.; Perslev, M.; Rezaiifar, R.; Rippel, O.; Sarasua, I.; Shen, W.; Son, J.; Wachinger, C.; Wang, L.; Wang, Y.; Xia, Y.; Xu, D.; Xu, Z.; Zheng, Y.; Simpson, A. L.; Maier-Hein, L.; and Cardoso, M. J. 2021. The Medical Segmentation Decathlon. arXiv:2106.05735.

Bidgood Jr., W. D.; Horii, S. C.; Prior, F. W.; and Van Syckle, D. E. 1997. Understanding and using DICOM, the data interchange standard for biomedical imaging. *Journal of the American Medical Informatics Association : JAMIA*, 4(3): 199–212. 9147339[pmid].

Blackledge, M. D.; Collins, D. J.; Koh, D.-M.; and Leach, M. O. 2016. Rapid development of image analysis research tools: bridging the gap between researcher and clinician with pyOsiriX. *Computers in biology and medicine*, 69: 203–212.

Chen, W.; Chen, X.; Zhang, J.; and Huang, K. 2017. Beyond triplet loss: a deep quadruplet network for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 403–412.

Cheng, X.; Zhang, L.; and Zheng, Y. 2018. Deep similarity learning for multimodal medical images. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 6(3): 248–252.

Clark, K.; Vendt, B.; Smith, K.; Freymann, J.; Kirby, J.; Koppel, P.; Moore, S.; Phillips, S.; Maffitt, D.; Pringle, M.; et al. 2013. The Cancer Imaging Archive (TCIA): maintaining and operating a public information repository. *Journal of digital imaging*, 26(6): 1045–1057.

Consortium, M. 2020. MONAI: Medical Open Network for AI.

Dikici, E.; Bigelow, M.; Prevedello, L. M.; White, R. D.; and Erdal, B. S. 2020. Integrating AI into radiology workflow: levels of research, production, and feedback maturity. *Journal of Medical Imaging*, 7(1): 016502.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv:2010.11929.

Eisenhauer, E. A.; Therasse, P.; Bogaerts, J.; Schwartz, L. H.; Sargent, D.; Ford, R.; Dancey, J.; Arbuck, S.; Gwyther, S.; Mooney, M.; et al. 2009. New response evaluation criteria in solid tumours: revised RECIST guideline (version 1.1). *European journal of cancer*, 45(2): 228–247.

European Society of Radiology (ESR). 2019. What the radiologist should know about artificial intelligence–an ESR white paper. *Insights into imaging*, 10: 1–8.

Gibaud, B. 2008. The DICOM Standard: A Brief Overview. In Lemoigne, Y.; and Caner, A., eds., *Molecular Imaging: Computer Reconstruction and Practice*, 229–238. Dordrecht: Springer Netherlands. ISBN 978-1-4020-8752-3.

Gillies, R. J.; Kinahan, P. E.; and Hricak, H. 2016. Radiomics: images are more than pictures, they are data. *Radiology*, 278(2): 563–577.

Hatamizadeh, A.; Tang, Y.; Nath, V.; Yang, D.; Myronenko, A.; Landman, B.; Roth, H.; and Xu, D. 2021. UNETR: Transformers for 3D Medical Image Segmentation. arXiv:2103.10504.

Hoffer, E.; and Ailon, N. 2015. Deep metric learning using triplet network. In *International workshop on similarity-based pattern recognition*, 84–92. Springer.

Jiang, F.; Jiang, Y.; Zhi, H.; Dong, Y.; Li, H.; Ma, S.; Wang, Y.; Dong, Q.; Shen, H.; and Wang, Y. 2017. Artificial intelligence in healthcare: past, present and future. *Stroke and vascular neurology*, 2(4).

Kang, S. K.; Mali, R. D.; Prabhu, V.; Ferket, B. S.; and Loeb, S. 2021. Active Surveillance Strategies for Low-Grade Prostate Cancer: Comparative Benefits and Cost-effectiveness. *Radiology*, 300(3): 594–604.

Kansagra, A. P.; Liu, K.; and John-Paul, J. Y. 2016. Disruption of radiologist workflow. *Current problems in diagnostic radiology*, 45(2): 101–106.

Koch, G.; Zemel, R.; Salakhutdinov, R.; et al. 2015. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille.

Kotter, E.; and Ranschaert, E. 2021. Challenges and solutions for introducing artificial intelligence (AI) in daily clinical workflow.

Kumar, V.; Gu, Y.; Basu, S.; Berglund, A.; Eschrich, S. A.; Schabath, M. B.; Forster, K.; Aerts, H. J.; Dekker, A.; Fenstermacher, D.; et al. 2012. Radiomics: the process and the challenges. *Magnetic resonance imaging*, 30(9): 1234–1248.

Panch, T.; Mattie, H.; and Celi, L. A. 2019. The "inconvenient truth" about AI in healthcare. *NPJ digital medicine*, 2(1): 1–3.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. 8024–8035.

Pichler, B. J.; Judenhofer, M. S.; and Pfannenberg, C. 2008. *Multimodal Imaging Approaches: PET/CT and PET/MRI*. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-540-72718-7.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. arXiv:1505.04597.

Segmentation, M.-O. 2015. Multi-Atlas Labeling Beyond the Cranial Vault - Workshop and Challenge. https://www.synapse.org/#!Synapse:syn3193805/wiki/217752. Accessed: 2021-09-29.

Subbaswamy, A.; and Saria, S. 2020. From development to deployment: dataset shift, causality, and shift-stable models in health AI. *Biostatistics*, 21(2): 345–352.

Tang, A.; Tam, R.; Cadrin-Chênevert, A.; Guest, W.; Chong, J.; Barfett, J.; Chepelev, L.; Cairns, R.; Mitchell, J. R.; Cicero, M. D.; et al. 2018. Canadian Association of Radiologists white paper on artificial intelligence in radiology. *Canadian Association of Radiologists Journal*, 69(2): 120–135.

Vallières, M.; Freeman, C. R.; Skamene, S. R.; and Naqa, I. E. 2015. A radiomics model from joint FDG-PET and MRI texture features for the prediction of lung metastases in soft-tissue sarcomas of the extremities. *Physics in Medicine and Biology*, 60(14): 5471–5496.

van Ufford, H. M. Q.; Kwee, T. C.; Beek, F. J.; van Leeuwen, M. S.; Takahara, T.; Fijnheer, R.; Nievelstein, R. A.; and de Klerk, J. M. 2011. Newly diagnosed lymphoma: initial results with whole-body T1-weighted, STIR, and diffusion-weighted MRI compared with 18F-FDG PET/CT. *American Journal of Roentgenology*, 196(3): 662–669.

Yu, K.-H.; Beam, A. L.; and Kohane, I. S. 2018. Artificial intelligence in healthcare. *Nature biomedical engineering*, 2(10): 719–731.

Appendix

The Unified Modeling Language (UML) diagram of the current architecture and software design for OsiriXgrpc can be seen in Figure 9 in the Appendix. This is still subject to changes as we are currently working on finalizing the initial version.

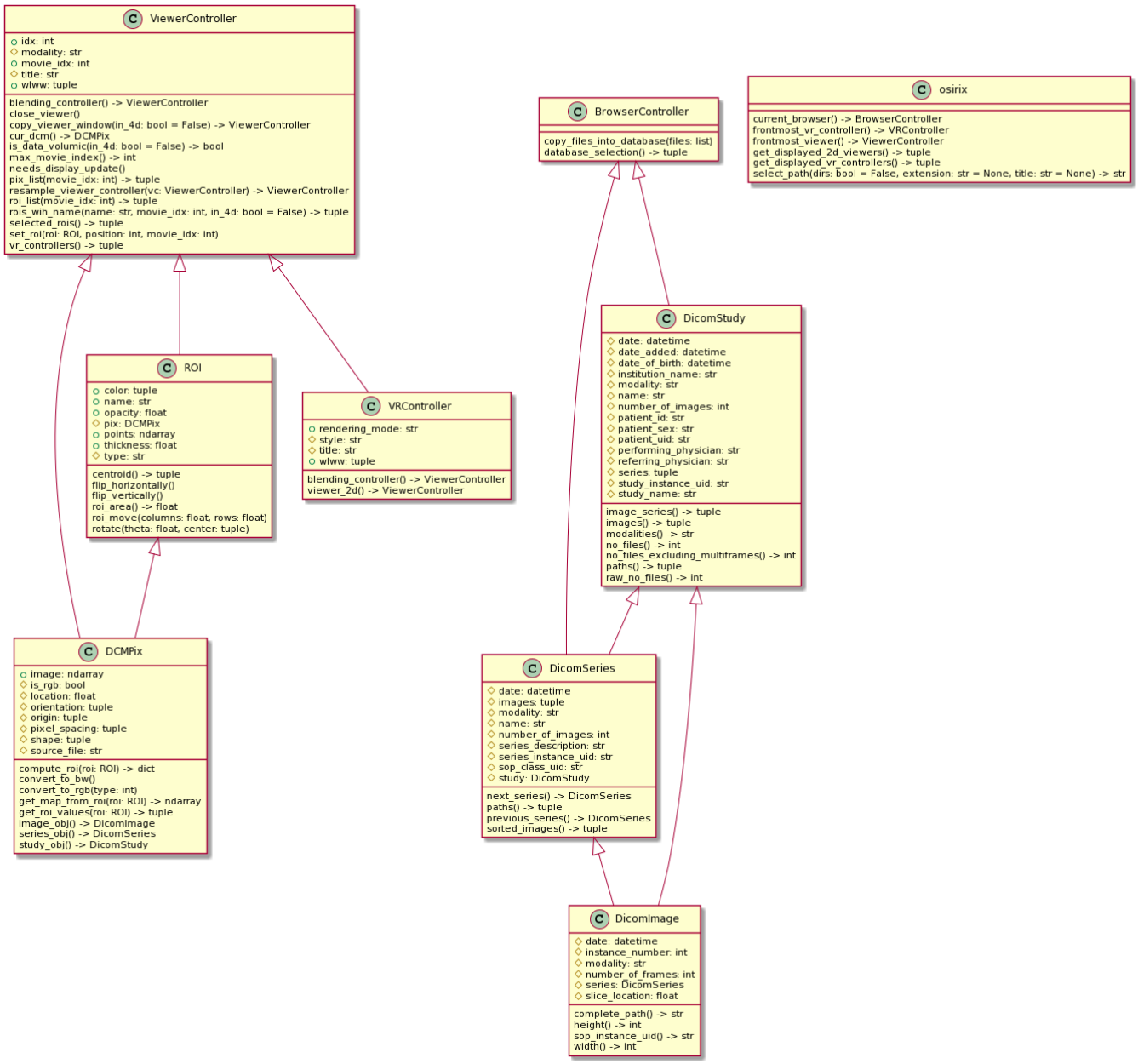

Figure 9: UML diagram for OsiriXgrpc software architecture.