Close to Real-Time Object Detection to Provide Synthetic Environments with Georeferenced Objects for Time-Critical Mission Planning

Arno Hollosi, Thomas Menzel-Berger
Blackshark.ai
AUSTRIA

ahollosi@blackshark.ai, tmenzel-berger@blackshark.ai

ABSTRACT

The ability to digest and iterate close to real-time geospatial data will be central to future modelling- and simulation-based (M&S) mission planning and rehearsal solutions in volatile, multi-domain scenarios. The ability to quickly integrate recently identified assets in M&S applications is a time-dependent bottleneck that can only be solved with the automated intake of timely real-world data.

This paper proposes an end-to-end geospatial pipeline that feeds current geo-data into a 3D digital twin environment for mission training and rehearsal. A fully automated solution based on a holistic global aerial data set (RGB, NIR, SAR) employs pre-trained ML-models to extract relevant geolocated 3D infrastructure and terrain data for a given mission area. A core feature of this technology is an integrated no-code, visual data labeling tool that enables non-AI/ML-trained personnel to apply ML analytics on current imagery to identify mission critical objects over large scale areas. These detected special interest objects are then automatically fed into the pipeline, georeferenced and visually represented in the synthetic rehearsal environment.

With this geospatial metaverse approach, synthetic training environments containing mission critical details can be created practically in real time. Future mission planning staffs will therefore have the ability to customize rehearsal simulations in an agile manner.

1.0 MODELLING AND SIMULATION IN TIME-CRITICAL SCENARIOS

2022 has challenged and changed the global security and defense landscape, especially in the northern hemisphere. Western democracies have been impacted by Russia’s invasion of Ukraine and its economic, social, and geopolitical implications. Additionally, the ever-looming tensions in the Indo-Pacific region remain a constant strategic concern. Both areas have once more shown the need for advanced geospatial planning and foresight; be it the status of humanitarian assets along NATO’s borders in Eastern Europe or an in-time assessment of the situation along a chain of islands in the South China Sea.

Geospatial information has demonstrated its paramount importance by enabling early warning signals and providing operational insights. In this paper we will present another area of application to harvest geospatial information for integrated training and simulation solutions. Governmental (i.e., intelligence communities) and commercial geospatial data providers (e.g., Maxar Technologies) deliver an abundance of close-to-real-time data for a chosen region of interest. For example, Maxar’s satellite fleet can deliver multiple revisit cycles per day if needed. The vast amount of generated geospatial data is a typical area of application for modern big data analytics backed by advanced machine learning models. Considering these technological advances, we will present an end-to-end geospatial platform to digest and analyze captured data (e.g., by drone or satellite) and output 3D-envoriments that can enable the next generation of modeling and simulation (M&S) solutions for mission critical planning and training.

A core feature of this platform is a no-code-based human-supervised interface to integrate new assets of interest into the creation of synthetic 3D environments for virtual mission planning and rehearsal. This patented innovation enables planners of operations to integrate non-predefined geo-assets into the training and detection cycles of the applied ML models. For example, if it is mission critical to find and secure water wells in an area of interest (AOI), a mission planner can add this information into the synthetic environment by simply tagging a hand full of examples in the provided interface for a given geographic region. The model will then take these new geo-labels, integrate them in its analysis cycles and feed representations of wells to the 3D synthetic training environment.

As this paper will show, the presented solution could enhance the plausibility and scope of M&S environments in the future. Most importantly, by accelerating the uptake cycles of geospatial analysis and the creation of authentic 3D environments, the process between data acquisition, automated analysis and mission planning could increase tremendously. This could enable and strengthen future digital twin ecosystems for security and defense through all domains (sea, air, land and cyber).

M&S has a long historic and incremental role in tactical and operational procedures. From early cardboard dioramas to current computer simulations, the need to pre-anticipate and forecast imponderabilia is central. Typically, the geographic terrain and other components of modern 3D environments are pre-modeled. Other relevant rehearsal information (e.g., the position of points, critical infrastructure, etc.) is either added manually or imported. This approach is limited to intelligence curated by personnel that is most likely off-site and by the long durations to bake these data layers into the synthetic mission scenarios. Considering these current limitations and temporal constraints, solutions for time-critical applications or even timely large-scale approaches are hard to achieve. This paper will present an innovation to fill the identified gap of synthetic 3D simulation augmentation by enabling a very rapid, automatic uptake of mission critical assets.

2.0 THE POWER OF MACHINE LEARNING APPLIED TO GEOSPATIAL DATA

Advances in machine learning enable fully automated analysis of satellite and aerial images for remote sensing purposes [1][2]. This allows large-scale information extraction (country-, continent-, or even planet-wide) which in turn empowers M&S applications to cover larger areas than ever before. Basically, machine learning (ML) is used to derive the base substrate for 3D digital twins. Given technologies like cloud computing [3][4], serverless computing [5] or edge computing [6], an M&S application is no longer limited to small-scale scenarios but can seamlessly integrate entire regions of interest.

Section 2.1 describes the ML-based analysis platform and Section 2.2 gives an example of features extracted from input imagery. Those features are then imported into and visualized by the 3D M&S environment. Section 2.3 discusses an efficient approach to build diverse geotypical 3D environments with limited resources in real time. Section 2.4 highlights the importance of timely updates, necessitating the improved labeling and training approach discussed in Section 3.0.

2.1 Architecture of a Geospatial Analysis Platform

Figure 2-1 (on next page) shows a prototypical architecture of a cloud-native geospatial analysis platform. At its core it consists of components for data management, workflow scheduling, training and inference, and user interfaces for data analysts and quality assurance as well as labeling. Note that in typical platform configurations, ML models are just one component of many orchestrated processes, functions, and data layers [7]. Using cloud-based services like blob storage, which are optimized for large workloads [8], container technology like Docker (https://docker.com/), orchestration platforms like Kubernetes (https://kubernetes.io/), and workflow engines like Flyte (https://flyte.org/) make it feasible to handle and process petabytes of geospatial data.

The core task of such platforms is operationalizing ML models [9] and assuring their lifecycle [10]. A typical ML life cycle contains data management (including typical extract/transform/load (ETL) steps), model learning (training), model verification, model deployment, and model execution [10]. The output of such processing steps (e.g., building footprints, vegetation masks, feature labels) is typically stored in standard formats for raster data (e.g., GeoTiff, or COG), vector data (e.g., GeoJSON, or GeoPackage), or in formats ready for consumption by client applications (e.g., USD, CDB, or 3D Tiles).

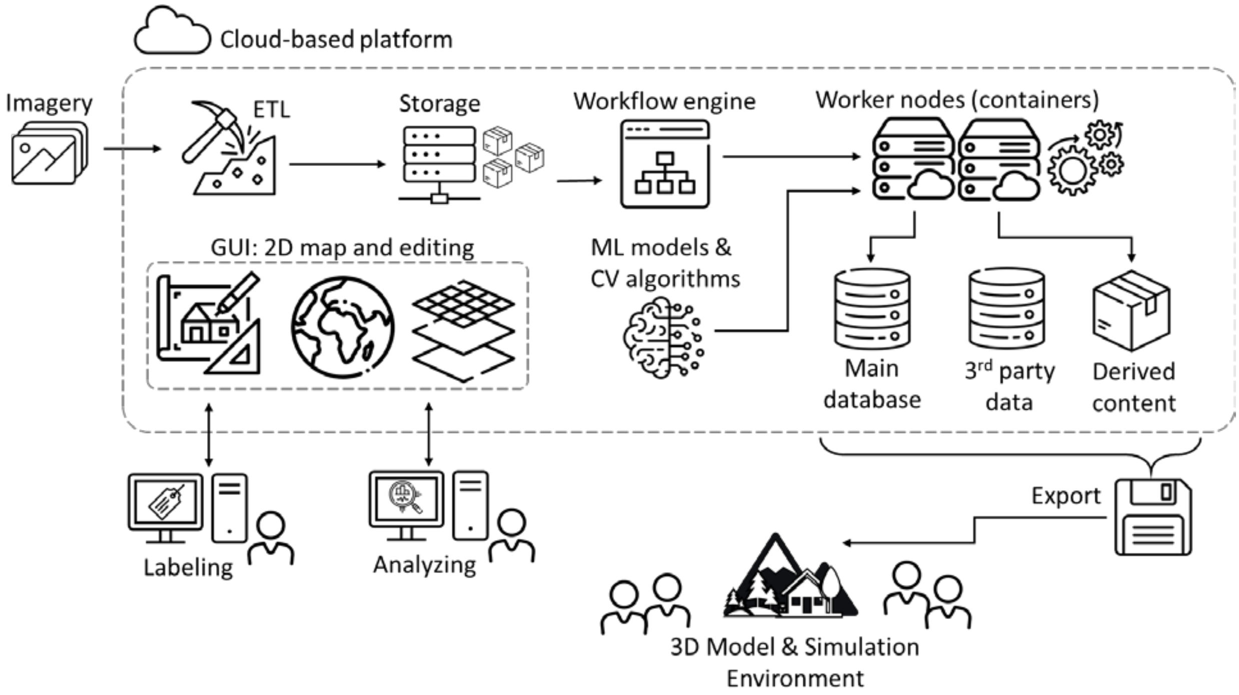

Figure 2-1: Cloud-based architecture of a typical geospatial analysis platform

2.2 Extracted Features

The presented geospatial platform contains a data management solution that is tailored to storing and versioning petabytes of georeferenced data and billions of geometrical features. A typical process comprises components for importing imagery (or other data types) into the platform, run machine learning models and other algorithms on this data, and consequently package the results for distribution to a 3D M&S environment. Figure 2-2 shows the general steps required for infrastructure detection in a densely populated area. The input image is loaded, roof formations are detected, and splits are analyzed for each respective building; the results are vectorized. Roofs and footprints as well as building heights are derived. The networks typically used for such tasks are either convolutional neural networks [11] or newer approaches like frame field learning [12].

This process has already been validated by commercial cases. For example, the presented framework processed the Iberian Peninsula (600,000 km^2) on a set of three dozen virtual machines in under 18h. In general, a common PC workstation with a single Graphics Processing Unit (GPU) can analyze several square kilometers per second with a local installment of the platform.

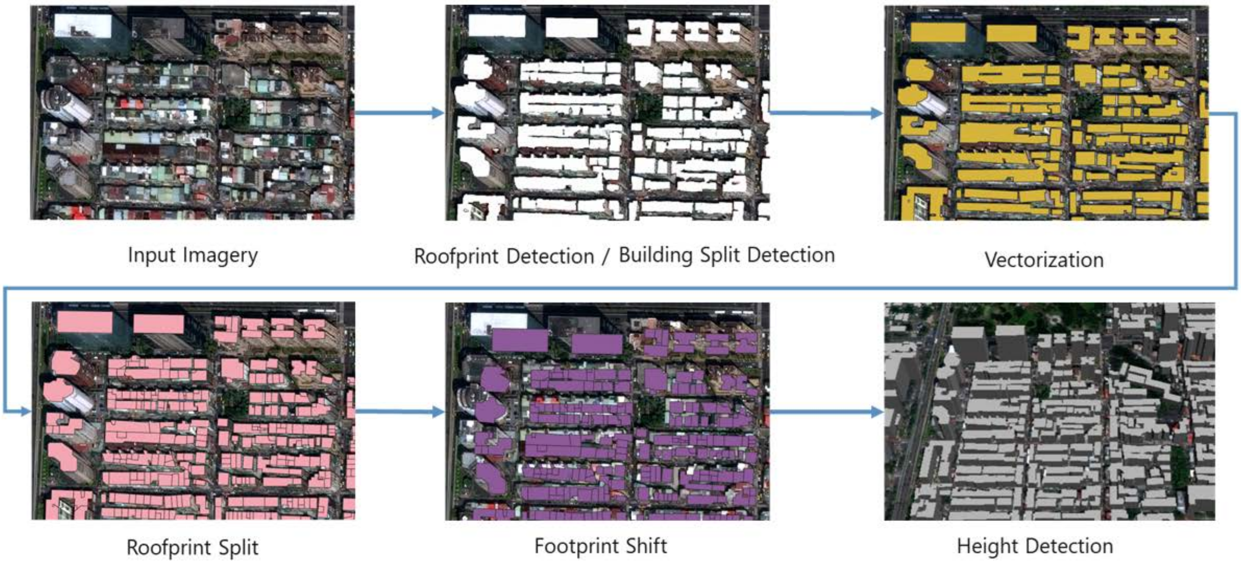

Figure 2-2: Extraction steps of building features

2.3 3D Visualization and Simulation

Once features have been extracted from the imagery, they can be used in M&S environments for visualization, analysis, planning, training, and other scenarios. The M&S application can be fully independent of the analysis platform described in Section 2.1 if both agree on a common format for the data transfer. In our case, we use a highly optimized format that can be streamed on demand, requires low bandwidth for transmission and yet produces rich and diverse results. The accompanying client software can be integrated into off-the-shelf 3D engines like Epic's Unreal Engine or Unity's 3D engine as well as propriety rendering engines and simulation applications.

Focusing on buildings and other infrastructure, one can either place ready-made 3D objects or procedurally generated objects (that are derived from the input data) into the simulation environment. One drawback of the first option is that typically there is only a limited number of 3D objects available. While those objects may have very high fidelity, they repeat quite often. Procedurally generated objects can generate an unlimited number of unique buildings and structures that fit the input data, e.g., buildings that match the detected footprint, building height, roof type, zoning, and other properties.

Our approach uses a patented procedural technology called Procedural Grammar Generator (PGG) [13] that takes in a diverse set of data sources and produces a matching digital twin in real-time. Other data sources – apart from detected features from input imagery – might include digital elevation models (DEM), road and rail networks, water bodies, and points of interest. The data representation is very efficient: for example, 1.5B buildings can be stored in about 30 GB (excluding textures) which enables storing a twin of the entire earth on mobile devices. The flexibility of PGG is achieved by a domain-specific language for describing building blueprints that are malleable and adaptable to the input data [14]. This enables a geotypical, diverse representation of regions across the entire planet. Figure 2-3 (next page) shows an example of three regions.

2.4 Timeliness of Updates

Depending on the use case, timeliness of updates is of utmost importance. The time from image capture to availability in the simulation environment should be as short as possible. This necessitates a fully automated approach.

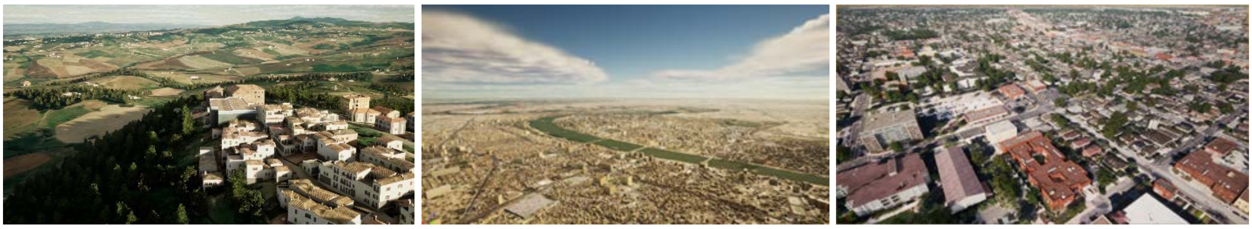

Figure 2-3: Geotypical rendering using PGG: Tuscany (Italy), Cairo (Egypt), and San Jose (California)

Given the architecture described above, one can see how new imagery is processed by each component in turn. The sequential nature of these steps might pose a problem when large swaths of data are ingested or updated. While geospatial data lends itself to easy parallelization, problems appear on the boundaries of the discrete work items that are scheduled for computation: later steps might need additional context from neighboring work items to function correctly. E.g., a vectorization step needs context for structures that span said item boundaries. With traditional scheduling frameworks, optimal parallelization either requires computing context areas multiple times, which is unnecessary overhead, or scheduling is reduced to simple parallelization within each step, but sequentially processing each step. This unnecessarily extends time to first simulation results and might not optimally use available compute resources. Our platform deploys a geospatially aware scheduling algorithm [15] that reduces turnaround times as it can take geospatial context dependencies into account when scheduling work items in parallel. Using this approach, first results are available without delay.

This still leaves a gap: only features where pre-trained ML models are available for detection can be used automatically. New features require a lengthy training process to generate reliable ML models. Ad hoc training of models by personnel in the field is not an option. In the next section, we are going to describe an approach for near real-time training for timely updates.

3.0 NEAR REAL-TIME TRAINING AND DETECTION

The methods and procedures outlined in Section 2.0 presume that machine learning models have already been trained on the desired detection classes and features. While efficient models will be available ahead of time (e.g., models for common infrastructure and vehicles), novel situations might require models for new classes to be trained ad hoc. The traditional approach would be to first generate a set of training data by a group of data analysts, then train the model, evaluate its quality, and eventually deploy it in the field. A process that potentially involves many people and takes considerable time.

3.1 Instant Feedback Labeling and Training

The presented solution utilizes a technique called *Live Labeling*. This approach contains a no-code, visual data labeling tool that enables non-AI/ML-trained personnel to apply ML analytics to novel data requirements in a fraction of the time of traditional state-of-the-art methods. With its short iteration time, its reduced need for personnel, and its integration into a 3D simulation environment, it is well suited for time-critical applications and scenarios. Figure 3-1 depicts the time and resources efficiency of this novel approach in comparison to a classic labeling approach to create geo-labels. Live Labeling outperforms classic approaches in the time spent per label as well as the cumulative number of created labels in the same duration.

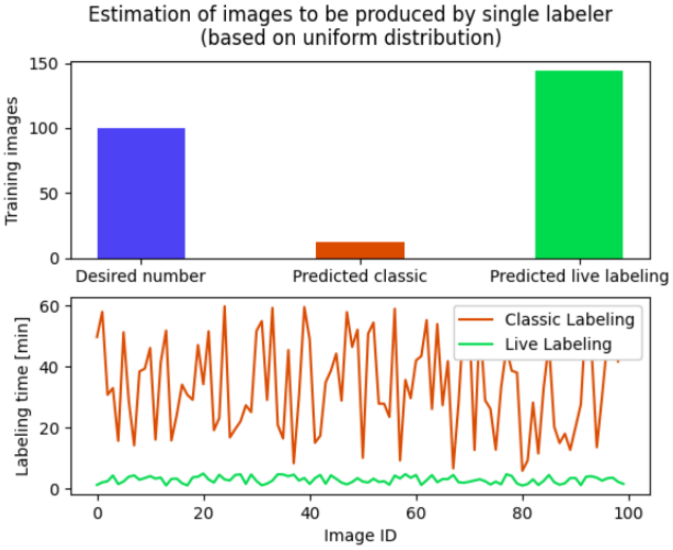

Figure 3-1: Speed and output comparison of classic vs. Live Labeling approaches

The visual interface of Live Labeling is shown in Figure 3-2. With a few strokes, an analyst identifies objects of interest (left-hand side). On the right-hand side the current inference of the model is shown. An analyst can zoom and pan the image, quickly identifying false positives and negatives and simply adding additional labels to correct those errors. Within minutes, the model achieves a high degree of accuracy.

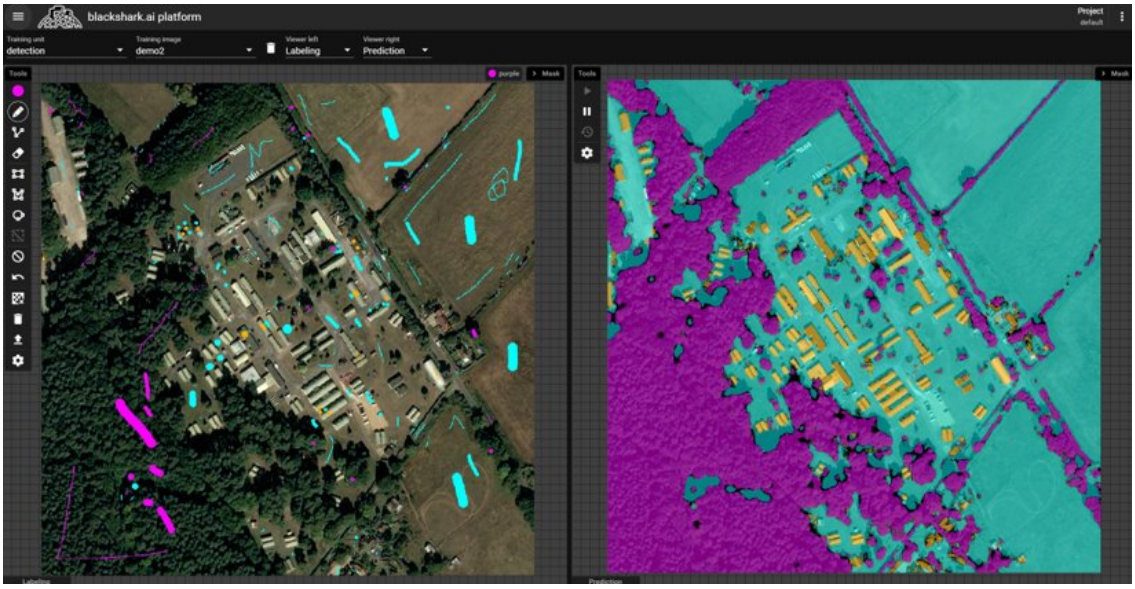
Training unit: detection
Training image: demo2
Viewer left: Labeling
Viewer right: Prediction

Figure 3-2: UI of Live Labeling in action; left the user’s annotation space, right the real-time prediction mask by the model under training

The interface is intuitive and due to its immediate visual feedback, it is easy to learn and understand. ML models are no longer a black box since data labelers see how each label improves (or worsens) the model’s detection accuracy. Hence, training new or additional personnel for this labelling tool can be done in a timespan of some hours to days, depending on the detection class, image quality, desired detection quality, and previous domain knowledge of the trainee. In our experience it is possible to onboard new data analysts in less than two days. Together with the speedup of labeling, this ease of use enables ad hoc training of people as required by the use case. Smaller labeling teams also mean that it is easier to form a team and find people who have the necessary security clearance.

3.2 Live Labeling Architecture

Live Labeling achieves its immediate feedback (new training epochs within seconds) by splitting the model into two training streams, as shown in Figure 3-3: one master/global model, which is continuously trained on all training data available, and one iterative/local model that is used for the visualization. The iterative model only takes the current training image as input and has a very high learning rate, hence magnifying the impact of every label made by the analyst. On the other hand, the master/global model uses a more conservative learning rate, eventually producing higher quality detections than the iterative model. Once quality criteria are satisfied, the master model can be run on large-scale areas from inside the same interface.

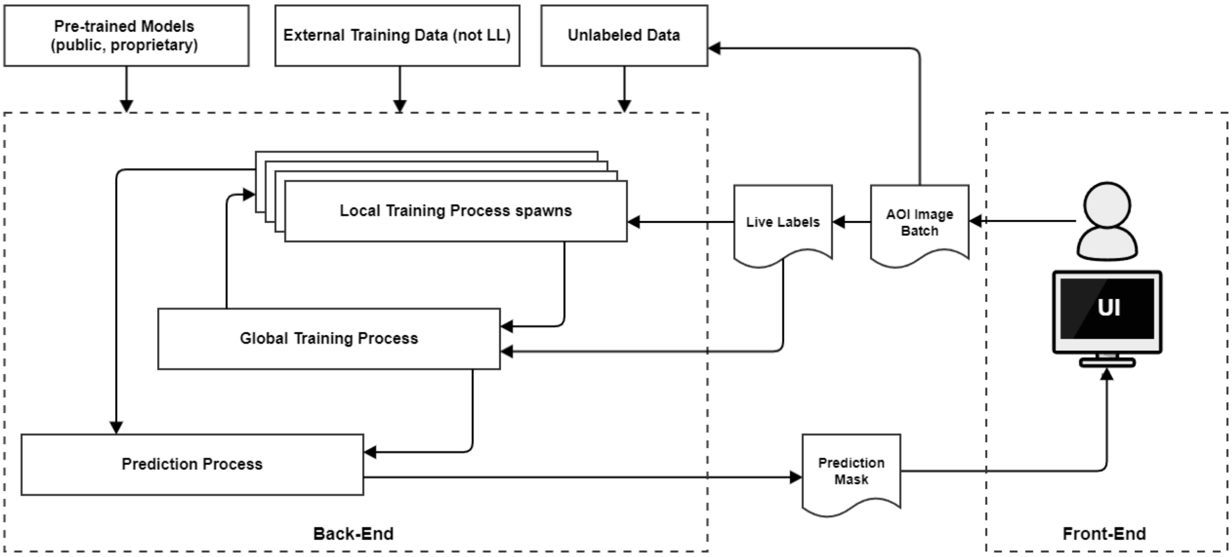

Figure 3-3 High-level schematic of the machine learning concept [16]

3.3 Example Scenario Using Close to Real-time Object Detection

Live Labeling can run on off-the-shelf hardware with consumer-grade GPUs. As such it can be used by personnel in the field, off-grid without network connection to the cloud. An example use case is depicted in Figure 3-4 (next page) in which local up-to-date imagery is captured by drones. This imagery is imported and one or two data analysist start labeling for a new detection class, e.g., transportation trucks. Within short time the first trucks and false positives are labeled. From there, Live Labeling guides the labeling process – by presenting additional uncertain results for labeling – and runs inference on ever larger areas, eventually covering the entire data.

The position of all detected trucks is then forwarded to the 3D simulation environment and ready-made 3D objects are placed at the given positions in the digital twin arena, as seen in Figure 3-5. In the 3D environment, one can visualize the count and distribution of the transportation trucks and, e.g., simulate their predicted route or obstacles in their path. The resources necessary for such a scenario fit inside a single server with enough disk capacity to store the input imagery and housing a handful of GPUs. Analysts can either use PC workstations or laptops for data entry as well as the 3D M&S environment.

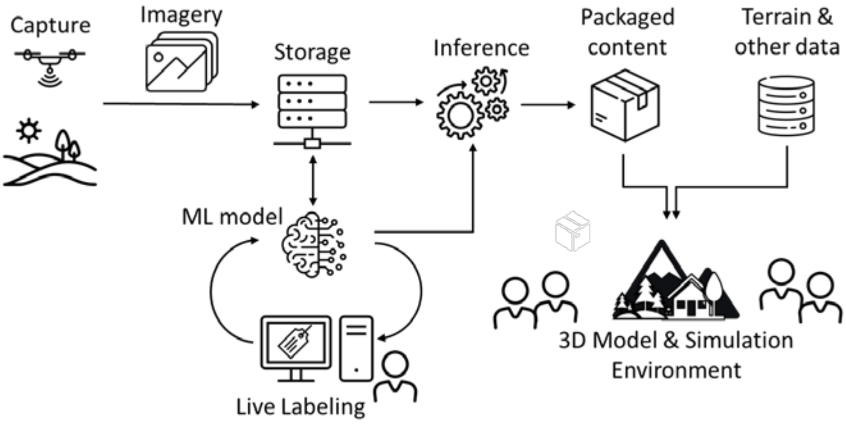

Figure 3-4: End-to-end use case for near real-time training, object detection, and simulation

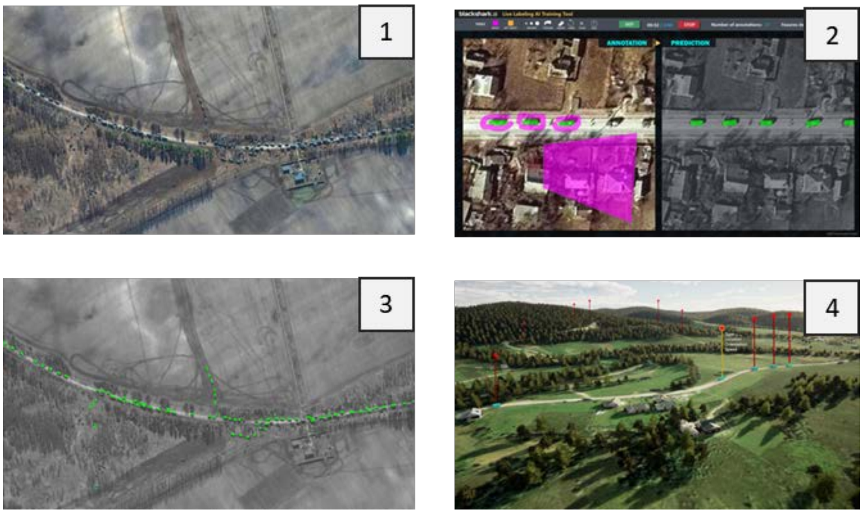

Figure 3-5: Data and activity of the end-to-end use case: (1) input imagery, (2) labeling trucks, (3) inference on imagery, (4) visualization in M&S application

4.0 FUTURE WORK

The described novel approach in Section 3.0 has been successfully demonstrated in several business cases. Live Labeling performs well when applied to scenarios where the segmentation requirements do not demand pixel-precise boundaries between labeling classes. Currently, object detection and segmentation with instance detection is a separate processing step independent of Live Labeling. Future work will therefore focus on integrating and enhancing the object detection capabilities of the Live Labeling tool to provide a wider range of potential use cases. The user interface of the tool is also constantly improved for better feedback, user experience, and performance.

In general, the proposed geospatial end-to-end framework needs further generalization for integration into multi-domain environments. For now, the pre-trained ML models are specialized on earth-bound geo-assets. Multi-domain simulations will require a more holistic detection capability to also cover e.g., seaborn assets.

In addition, to integrate with existing M&S environments, export data formats such as CDB [17] or 3D Tiles [18] need to be supported. The challenge there is that our custom approach (using PGG) does not retain all its properties when 3D objects are baked into other formats. E.g., the size of the data set explodes hundred-fold, as those formats store the 3D mesh itself instead of meta properties like PGG does. Also, some features like client-side coloring and layered rendering cannot currently be represented in those formats leading to poorer visual quality. We are working actively on solutions to these challenges.

5.0 CONCLUSION

This paper has illustrated the potential power of a fully integrated geospatial end-to-end framework for future Modelling and Simulation as a Service (MSaaS) approaches. With the right base conditions (compute resources, data acquisition & access, system integration) timely mission planning scenarios can be created and delivered on the fly for all defense and other domains. By harnessing the described ML-based human-involved approach, future training environments will become closer to reality and customizable for non-AI-experts. The acceleration of the data acquisition, analysis and mission planning cycle is thereby a key advantage.

The discussed technology could also have a broader impact that goes beyond the immediate use-case highlighted in this paper. Future digital twin applications for defense purposes will be generated more quickly creating more informed mission scenarios by making the complexity of fully integrated systems comprehensible and manageable. Ultimately, switching between live missions and training environments could become so seamless that it will blur the boundaries between real-world operations and synthetic mission scenarios.

6.0 ACKNOWLEDGEMENTS

Daniel Lahm (technical writer) and Stefan Habenschuss (head of machine learning) at Blackshark.ai helped with the concept and the revision of this paper. Icons in Figure 2-1 and Figure 3-4 are from Flaticon.com.

7.0 REFERENCES

[1] Neupane, B., Horanont, T., & Aryal, J. (2021). Deep Learning-Based Semantic Segmentation of Urban Features in Satellite Images: A Review and Meta-Analysis. *Remote Sensing*, *13*(4), 808. MDPI AG. Retrieved from http://dx.doi.org/10.3390/rs13040808

[2] Asokan, A., & Anitha, J. (2019). Machine learning based image processing techniques for satellite image analysis-a survey. In *2019 International conference on machine learning, big data, cloud and parallel computing (COMITCon)* (pp. 119-124). IEEE.

[3] Surbiryala, J., & Rong, C. (2019, August). Cloud computing: History and overview. In *2019 IEEE Cloud Summit* (pp. 1-7). IEEE.

[4] Alghofaili, Y., Albattah, A., Alrajeh, N., Rassam, M. A., & Al-rimy, B. A. S. (2021). Secure Cloud Infrastructure: A Survey on Issues, Current Solutions, and Open Challenges. *Applied Sciences*, *11*(19), 9005.

[5] Schleier-Smith, J., Sreekanti, V., Khandelwal, A., Carreira, J., Yadwadkar, N. J., Popa, R. A., ... & Patterson, D. A. (2021). What serverless computing is and should become: The next phase of cloud computing. *Communications of the ACM*, *64*(5), 76-84.

[6] Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, *3*(5), 637-646.

[7] Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., ... & Dennison, D. (2015). Hidden technical debt in machine learning systems. *Advances in Neural Information Processing Systems*, *28*.

[8] Sun, X., Li, K., Ren, Y., Lin, J., Ren, Z., Feng, S., ... & Qi, Z. (2021). Client layer becomes bottleneck: workload analysis of an ultra-large-scale cloud storage system. In *Proceedings of the 14th IEEE/ACM International Conference on Utility and Cloud Computing Companion* (pp. 1-6).

[9] Kolltveit, A. B., & Li, J. (2022, May). Operationalizing Machine Learning Models – A Systematic Literature Review. In *2022 IEEE/ACM 1st International Workshop on Software Engineering for Responsible Artificial Intelligence (SE4RAI)* (pp. 1-8). IEEE.

[10] Ashmore, R., Calinescu, R., & Paterson, C. (2021). Assuring the machine learning lifecycle: Desiderata, methods, and challenges. *ACM Computing Surveys (CSUR), 54*(5), 1-39.

[11] Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. In *2017 international conference on engineering and technology (ICET)* (pp. 1-6). IEEE.

[12] Girard, N., Smirnov, D., Solomon, J., & Tarabalka, Y. (2021). Polygonal building extraction by frame field learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 5891-5900).

[13] Thaller, W., Richter-Trummer, T., Putz, M., & Maierhofer, R. (2020). *U.S. Patent No. 10,636,209*. Washington, DC: U.S. Patent and Trademark Office.

[14] Hollosi, A., Menzel-Berger, T., Walter, H., Lahm, D. (2022) Automated 3D Building Generation at Global Scale Based on Satellite Imagery. In *Proceedings of Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC).*

[15] Hollosi, A., & Schlager, F. (2022). U.S. Patent No. 11,372,687. Washington, DC: U.S. Patent and Trademark Office.

[16] Habenschuss, S., Hollosi, A., Kuksa, P., & Presenhuber, M. (2021). *U.S. Patent No. 11,049,044*. Washington, DC: U.S. Patent and Trademark Office.

[17] Reed, C. (Ed.). (2021). *Volume 1: OGC CDB Core Standard: Model and Physical Data Store Structure*. Open Geospatial Consortium. Retrieved May 11, 2022 from http://www.opengis.net/doc/IS/CDB-core/1.2

[18] Cozzi, P., Lilley, S., Gabby, G. (Eds.). (2018). *3D Tiles Format Specification. Version 1.0.* Cesium GS, Inc. Retrieved May 11, 2022, from https://github.com/CesiumGS/3d-tiles/tree/main/specification