A DESIGN PROPOSAL FOR MACHINE-ASSISTED PHOTO-INTERPRETATION IN NATIONAL FOREST INVENTORIES

Vladislav TIFTILOV[3], Ciprian DOBRE[4], Olivier BOURIAUD[5]

Abstract. *National forest inventories track how forest area changes over time. The resulting figures become official statistics, so they must be accurate. To produce them, a grid of evenly spaced points is placed over the country. An expert looks at aerial images and classifies each point by hand. This work is slow and expensive. This paper proposes a design for a system that supports this work using satellite images, and states the requirements it must meet. A classifier decides the points it is sure about and passes the uncertain ones to an expert. We ran two comparisons on Sentinel-2 images, changing one design choice at a time. What matters most is reusing a neural network already trained on everyday photographs. Such a network, with its weights kept fixed, reaches 92.8% accuracy on a benchmark. The results suggest that machine assistance is practical here.*

Keywords: machine-assisted photo-interpretation, national forest inventory, land-use classification, transfer learning, system requirements

DOI 10.56082/annalsarsciinfo.2026.2.22

1. Introduction

National forest inventories are a common way to keep track of forest areas in a country. In Romania the inventory runs in cycles, and a single cycle takes several years [1]. To mark the forest area, a grid of points is placed over the country, and an operator classifies each point by looking at aerial images. The next step is to go to the field and measure the trees for a sample of locations, which is used to estimate the volume of timber. This process is slow, expensive and varies between operators. It is important to have accurate results because they become official statistics.

The challenging aspect of this workflow is the scale. The Romanian grid has almost one million points [1], and each point must be classified manually. Automating it is not enough because no model can guarantee perfect accuracy, which is why we want to rely on both an automation and an operator to improve the results. Existing systems classify remote sensing imagery with high accuracy [2], but they rarely

[3] National University of Science and Technology POLITEHNICA Bucharest, Romania, vtiftilov@stud.acs.upb.ro
[4] Professor, Faculty of Automatic Control and Computers, National University of Science and Technology POLITEHNICA Bucharest, Romania, ciprian.dobre@cs.pub.ro
[5] Professor, Faculty of Forestry, Stefan cel Mare University of Suceava, Romania, obouriaud@usm.ro

measure their own uncertainty. This is why we propose a system that classifies the points it is confident about and sends the rest to an operator.

This article is a design proposal, organised as follows: Section 2 reviews existing automated systems; Section 3 states the requirements; Section 4 presents the design decisions and the experiments behind them; Section 5 explains supervised classification and conformal prediction; Section 6 describes the proposed architecture; and Section 7 lists the open problems.

2. Related Work

Existing literature describes several systems that have the same goals as our proposal. Yet none of them meets all the requirements we set for a supervised classification system that routes uncertain points to a human operator.

The closest work to our proposal is Moraes et al. [3]. They train their model on the Portuguese inventory. Similar to Romania, the first phase uses a grid of points spaced 500 m apart. They use convolutional networks to label every pixel of an image from Sentinel-2 imagery. To solve the cloud problem they use images of the same plot from multiple dates. The difference is that the output of their system is a per-pixel land cover map, not a decision at each inventory point. They report an accuracy of 71.3% on the Portuguese inventory, but the results are not comparable because of different classification units and the classes differ.

The next work worth mentioning is Martinez-Sanchez et al. [4]. The LUCAS survey is a European survey that collects information about land use by physically travelling to the location and taking photos. To automate the survey, they use a convolutional network to label every pixel of a photo taken by the surveyor. Similar to our proposal, they use a pretrained network and then apply a random forest over its output to predict the survey class. They report an F1 score of 0.89, but this result is not that relevant to our work because we use satellite images.

The last work we mention is Sylvain et al. [5]. They use aerial imagery of 30 cm resolution per pixel to map tree species. They also use convolutional networks for classification, but they train three different network on three different datasets, which gives nine models in total. They combine all nine and use the disagreement between the models to measure uncertainty. This is the only system that measures its own uncertainty, but it does not use it to route points to a human operator.

In conclusion, there are systems that automate part of the photo-interpretation task, but none of them tries to route the uncertain points to a human operator based on an uncertainty score. This is the main difference between our proposal and the existing literature.

3. Requirements

In this section we present the requirements for a machine-assisted photo-interpretation system that supports national forest inventories. The requirements are based on the current workflow. The system must classify most of the grid points automatically and send the uncertain ones to a human operator for review. The labels it makes must come with a guarantee on how often the automatic decision is wrong, which is why we use conformal prediction.

3.1. Operational Context

The Romanian national forest inventory happens in two phases [1]. During the first phase, a grid of points is laid over the country. The space between the points is 500 m, and there are almost 1 million points in total. An operator looks at aerial images for each point and classifies it as either forest or not. These labels are not published on their own. They give the share of the country that is forest, so the per-hectare field measurements can be scaled up to the whole country.

The second phase defines a network of 125,000 field plots, spaced 2 km apart in lowlands and 4 km apart in hills and mountains [1]. Crews measure the trees in the plots they visit, and the measurements are used to estimate the volume of timber.

A separate photo-interpretation decides which of those plots the crews should visit. A crew visits a plot if the interpreter sees forest vegetation there, but also if the interpreter cannot be sure of it. The inventory therefore already spends field effort to resolve doubtful cases, and this is why we want to automate it.

The workload is fixed and large: almost 1 million decisions per cycle, none of which can be skipped. And forest is rare in the lowlands, at 6% of the points, but common in hills and mountains, at 45% [1], so a single national model or calibration set would be dominated by the mountain case. These two properties, size and class imbalance, are the most important when defining the requirements for the system.

3.2. Functional Requirements

Functional requirements describe what the system must do. The requirements are:

- **Classify every grid point**. For each grid point the system returns every class that could apply to it, using only the imagery for that point.
- **Guarantee the set contains the true class**. The set must hold the correct class in at least 90% of points, a rate fixed before the run.
- **Send unresolved points to an operator**. A set with one class is accepted as the label. A set with more than one class is unresolved, so the point goes to an operator.

- **The operator decision is final**. An operator label replaces the model output. A later prediction may not overwrite it.

3.3. Non-Functional Requirements

The non-functional requirements set the limits the system has to work within:

- **Throughput**. The system labels the full grid of 950,000 points within one day. That is about 11 points per second, making it feasible to re-run the classifier after changing the model.
- **Review latency**. When an operator opens a point, the tile and the proposed classes appear in under a second. One second is the usual limit for an interface to feel immediate, so the operator's attention stays on the image rather than on the tool.
- **Operator budget**. At most 20% of the grid is sent for review, which is 190,000 points instead of the 950,000 interpreted by hand today. Setting the limit in advance allows the inventory to plan the review effort.
- **Bounded compute**. The full run fits on a single workstation with eight cores and 32 GB of memory, and needs under 50 GB of storage for the tiles and their embeddings. This allows the model to be run on a common office computer.
- **Traceability**. Every label is stored with the model version, the imagery date, the prediction set, and whether an operator intervened. Official statistics are built on these labels, so each one has to be explainable.

4. Design Decisions and Supporting Evidence

In this section we discuss the design decisions we made to meet the requirements. We start from the choice of imagery source, then we discuss the choice of model family and training strategy, and finally we discuss the choice of pretrained network.

4.1. Choice of Imagery Source

The Romanian NFI photo-interpretation currently uses aerial imagery [1]. Flying a plane over the whole country is expensive, and the campaigns depend on the schedule of the aircraft which it cannot control. A requirement for the imagery source is that it must cover the whole country, repeat at regular intervals, and be cheap enough to allow reprocessing the full grid at each campaign, or whenever the model changes. Satellite imagery meets these conditions. Sentinel-2 is distributed under the Copernicus free and open data policy [6]. Covering the whole country for one cloud-free date takes about 19 GB, so reprocessing the full grid is cheap.

The drawback of this choice is resolution. At 10 m per pixel, a grid cell of 500 m by 500 m is only 50 by 50 pixels, far coarser than the aerial imagery interpreters use today. It is still an open question whether this resolution is enough for the inventory. The experiments in this paper reach a high accuracy on EuroSAT [7], a benchmark built from Sentinel-2 imagery. But classes are different from the ones used in the inventory, so it does not prove the same accuracy holds for the inventory.

4.2. Evaluation Setup for the Supporting Experiments

To benchmark each design decision, we use EuroSAT [7], a public dataset of Sentinel-2 images. It contains 10 classes of land use and 27,000 images in total. We sample 300 images per class. We train the model on 210 images from each class to make sure that every class is equally represented in the training and test sets. The remaining 90 images per class are used for testing. We tune the model based only on the training images, therefore the test images are seen only once at test time. We never change more than one part of the model in a single comparison, which allows us to isolate the effects of each design decision.

The first limitation of using EuroSAT is that it uses a different list of classes than the inventory. The results therefore might differ when we switch to the same classes. The second is the size of the test set, 900 images measure accuracy only to about one point, therefore small differences between results are ignored.

4.3. Choice of Model Family

To determine the best model family, we ran a comparison on EuroSAT, summarised in Table 1 and Figure 1. The first attempt was a logistic regression on raw pixels, which reached 30.6% accuracy. The second attempt was a random forest on colour statistics, which reached 72.4% accuracy. Both of these approaches ignore visual features, like lines and circles, which makes it hard to recognise more complex shapes, like roads and trees.

The third attempt was a small convolutional network trained from scratch, which reached 78.8% accuracy. A convolutional network can learn visual features, but the dataset is too small to achieve higher accuracy.

The next attempt was to use transfer learning, which reuses a network already trained on ImageNet, which was built from over a million everyday photographs [8]. The features a network can learn from transfer are enough to work on tasks it never saw during training [9]. We also used ResNet-18, an eighteen-layer network, which reached 91.0%. Residual networks are efficient when the network has many layers because each layer learns a correction to the previous layer [10]. Finally, we fine-tuned the ResNet-18 network, which reached 94.3% accuracy. Fine-tuning means that the weights of the network are updated during training, which allows the network to learn features specific to the new dataset.

We choose the network with fixed weights. Fine-tuning improves accuracy, as shown in [9], but it would require processing all 950,000 tiles again. With fixed weights the result of each tile can be stored and reused, so improving the system later costs almost nothing.

Table 1. Model-family comparison on the same tiles and split. The difference between the best and worst model family is 64 points, which is far more than any later design choice.

Approach	**Accuracy (%)**	**Macro-F1**
Fine-tuned ResNet-18	94.3	94.3
Frozen ResNet-18 + linear classifier	91.0	91.0
Custom CNN trained from scratch	78.8	77.1
Random forest on colour features	72.4	71.8
Logistic regression on raw pixels	30.6	29.7

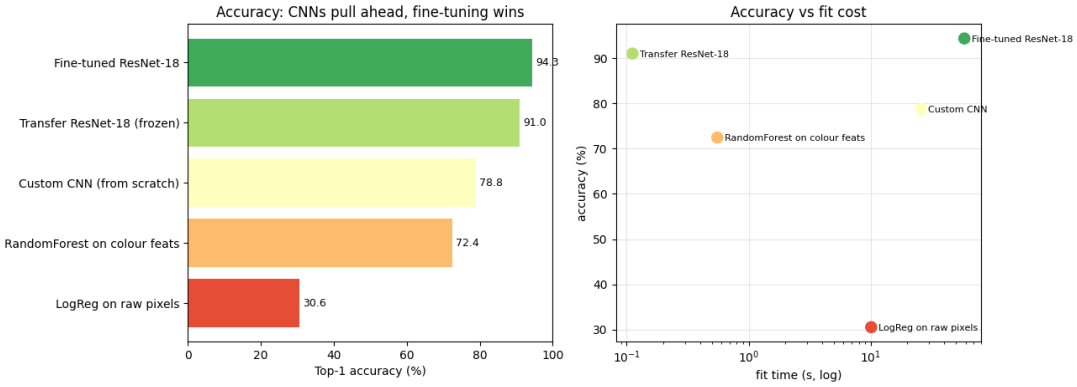

Fig. 1. Accuracy of five model families on the same task and split, from a raw-pixel linear model to a fully fine-tuned convolutional network.

4.4. Choice of Pretrained Network

The next tests were meant to determine which pretrained network to use, and the comparison is in Table 2 and Figure 2. The difference between the best and worst backbone is only 3.8 points, which is small compared to the 64 points spread between model families. We also discovered that the size of the network does not directly correlate with accuracy. AlexNet [11] produces the widest output, 9,216 numbers per tile, but it scores the lowest. MobileNetV3-Small [12] is one of the smallest networks, but it scores higher than ResNet-18 [10], which has twelve times more weights.

The conclusion is that the quality of the network matters more than its size. We choose EfficientNet-B0 [13], which reaches the best accuracy with only 4.0 million weights. Choosing a smaller network benefits the system because it reduces the time to process each tile, which is important when the system must process 950,000 tiles per run.

Table 2. Six pretrained networks, each with fixed weights and the same classifier on top. Model size varies significantly, but accuracy varies by less than four points.

Backbone	Feature dim.	Params (M)	Accuracy (%)
EfficientNet-B0 [13]	1280	4.0	92.8
ResNet-50 [10]	2048	23.5	91.6
MobileNetV3-Small [12]	576	0.9	91.3
ResNet-18 [10]	512	11.2	91.1
SqueezeNet 1.1 [14]	512	0.7	89.9
AlexNet [11]	9216	2.5	89.0

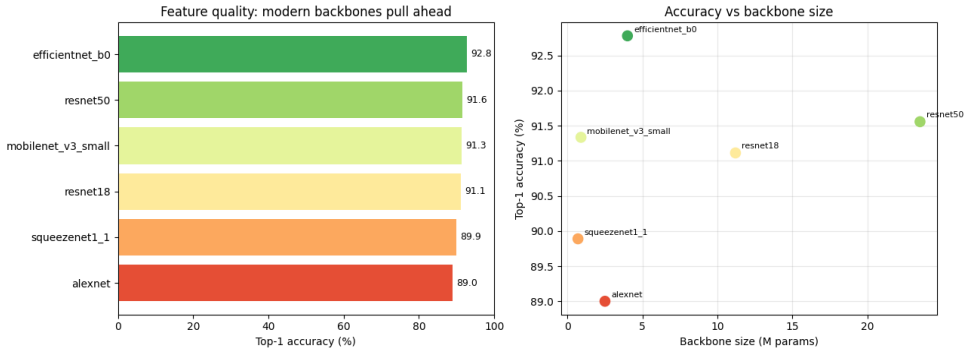

Fig. 2. Accuracy of the same six networks. The spread is narrow, so any of these networks would do.

5. Supervised Classification and Conformal Prediction

Supervised classification means that the model is trained on already labelled tiles. This approach is suitable for NFI because there are already three previous cycles of photo-interpretation we can use as training data.

An alternative to supervised classification is active learning, which routes to the operator the points that the model would learn most from. This approach is less suitable for our use-case because we already have the labels from previous cycles, and because the operator has to resolve the hard tiles whether or not the model

would learn from them. Another alternative is to do fully manual interpretation, which is the current workflow. This approach is also less preferable because it is slow, expensive and inconsistent between operators.

When the model is trained, it produces a score for each class, but the highest score does not always mean the class is right. Conformal prediction is a method that turns the scores into a set of classes, which is guaranteed to contain the correct class with a pre-specified probability [15]. For this, some labelled tiles are held back for calibration, and they are used to find how high a score must be for the set to reach that probability on new tiles. When conformal prediction gives a single class for a tile, the label is accepted. When it gives more than one class, the tile is sent to an operator for review. The operator can then choose the correct class, which is stored as the final label. This approach of routing the uncertain tiles to an operator is supported by Cresswell et al. [16], who found that people decide more accurately when shown a conformal set than when shown a fixed number of options.

6. Proposed Architecture

Figure 3 shows the proposed architecture. The current design is a proof of concept, therefore it does not include redundancy or replication.

The pipeline starts from downloading the Sentinel-2 images from the public archive. Only the 500 m window around each grid point is read, and the window is cached on disk so the same tile is never fetched twice. The tile is then passed through a pretrained network, which produces a vector of numbers, called an embedding. Because we use a pretrained network with fixed weights, the embeddings do not change over time, so they are cached on disk as well. Downloading and embedding are the most expensive steps, this is why it is important to cache them.

The next step is to run a logistic regression on the embedding, which produces a probability for each class. We use logistic regression because it returns a probability per class directly, which is a convenient conformal score. Conformal prediction then turns the probabilities into a set of classes, and the size of that set decides what happens next, as described in Section 5.

Every label is stored together with what produced it, as the traceability requirement mentions: the model version, the imagery date, the prediction set, and whether an operator intervened.

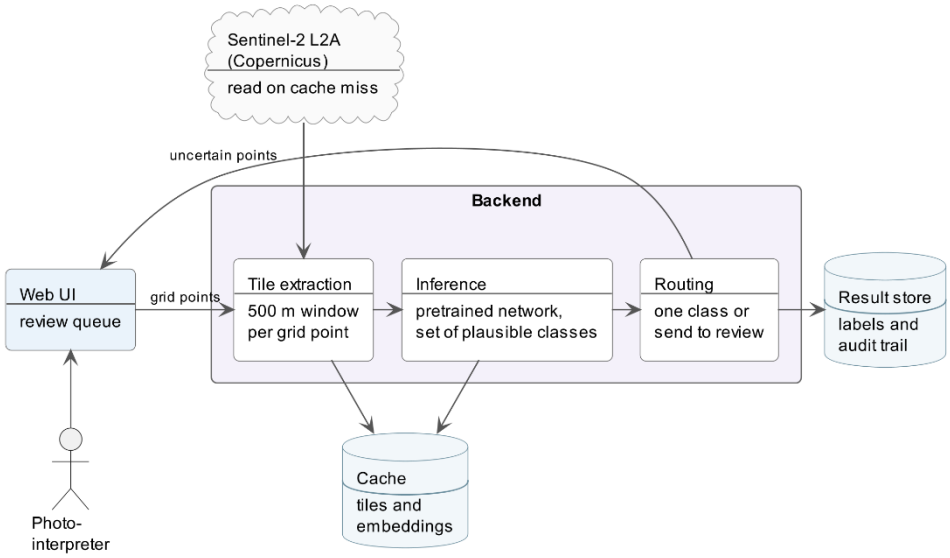

Fig. 3. Proposed architecture. A 500 m window is extracted around each grid point, a pretrained network and a classifier produce a set of classes, and routing accepts a single-class set or sends the point to an operator. Tiles and embeddings are cached, so each tile is downloaded and embedded once.

7. Limitations and Open Problems

While designing the system, we found several limitations and open problems that might affect the final implementation:

- **Image resolution**. Sentinel-2 images have a resolution of 10 m per pixel [6], which is lower than that of the aerial imagery used today. It is not clear whether this resolution is enough for the classes used in the inventory. One alternative is to use the same aerial imagery, another is to re-purpose the proposed system into a support tool for improving the existing workflow rather than replacing it.
- **Benchmark classes vs inventory classes**. For the design decisions we used EuroSAT, a benchmark dataset that uses a different set of classes than the inventory. We still have to confirm that the same models will work with the inventory classes.
- **Operator user experience**. The design aims to route to the operator less than 20% of the points. We have to confirm empirically that 20% is a reasonable limit for the operator to handle, and that the interface is fast enough for large numbers of points.

- **Clouds**. Satellite images may contain clouds, which makes it impossible to classify the points. We have to confirm that using images from multiple dates is manageable by the system since it requires downloading and embedding multiple tiles for the same point, and to decide what happens when no clear image exists for a point.

8. Conclusions

This article proposed a design for a machine-assisted photo-interpretation system for national forest inventories. The system classifies the points it is confident about and sends the rest to an operator. This is why we use conformal prediction, it allows the system to guarantee that the correct class is in the resulting set of classes at a rate chosen before the run.

Based on the experiments, we chose a convolutional network pretrained on ImageNet, with fixed weights, and a logistic regression on top. This setup allowed us to reach an accuracy of 92.8% on EuroSAT, a benchmark built from Sentinel-2 imagery. The design decisions were justified by the experiments, and the requirements were based on the current workflow.

There is still work to be done to confirm that the design works with the inventory's own imagery and classes, and that the operator can handle the review workload. We also plan to explore solutions to further improve the accuracy, such as anomaly detection and self-supervised pretraining.

R E F E R E N C E S

[1] O. Bouriaud, G. Marin, J.-C. Hervé, T. Riedel, and A. Lanz, *Estimation Methods in the Romanian National Forest Inventory*. Environmental Science, Engineering and Technology, New York: Nova Science Publishers, 2020.

[2] G. Cheng, X. Xie, J. Han, L. Guo, and G.-S. Xia, "Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 3735–3756, 2020.

[3] D. Moraes, M. L. Campagnolo, and M. Caetano, "A weakly supervised and self-supervised learning approach for semantic segmentation of land cover in satellite images with national forest inventory data," *Remote Sensing*, vol. 17, no. 4, p. 711, 2025.

[4] L. Martinez-Sanchez, L. See, M. Yordanov, A. Verhegghen, N. Elvekjaer, and D. Muraro, "Automatic classification of land cover from LUCAS in-situ landscape photos using semantic segmentation and a random forest model," *Environmental Modelling & Software*, vol. 172, p. 105931, 2024.

[5] J.-D. Sylvain, G. Drolet, É. Thiffault, and F. Anctil, "High-resolution mapping of tree species and associated uncertainty by combining aerial remote sensing data and convolutional neural networks ensemble," *International Journal of Applied Earth Observation and Geoinformation*, vol. 131, p. 103960, 2024.

[6] M. Drusch, U. D. Bello, S. Carlier, O. Colin, V. Fernandez, F. Gascon, B. Hoersch, C. Isola, P. Laberinti, P. Martimort, A. Meygret, F. Spoto, O. Sy, F. Marchese, and P. Bargellini, "Sentinel-2: ESA's optical high-resolution mission for GMES operational services," *Remote Sensing of Environment*, vol. 120, pp. 25–36, 2012.

[7] P. Helber, B. Bischke, A. Dengel, and D. Borth, "EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 7, pp. 2217–2226, 2019.

[8] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, *et al.*, "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[9] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Advances in Neural Information Processing Systems 27 (NIPS)*, pp. 3320–3328, 2014. arXiv:1411.1792.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017. AlexNet; originally published at NeurIPS 2012.

[12] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, *et al.*, "Searching for MobileNetV3," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1314–1324, 2019.

[13] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, vol. 97 of *Proceedings of Machine Learning Research*, pp. 6105–6114, PMLR, 2019.

[14] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and less than 0.5MB model size," *arXiv preprint arXiv:1602.07360*, 2016.

[15] A. N. Angelopoulos and S. Bates, "Conformal prediction: A gentle introduction," *Foundations and Trends in Machine Learning*, vol. 16, no. 4, pp. 494–591, 2023.

[16] J. C. Cresswell, Y. Sui, B. Kumar, and N. Vouitsis, "Conformal prediction sets improve human decision making," in *Proceedings of the 41st International Conference on Machine Learning*, vol. 235 of *Proceedings of Machine Learning Research*, pp. 9439–9457, PMLR, 2024..