ALLSKOG
EID AV SKOGEIERE

FINAL REPORT

BRASKOG FASE 2

Bærekraftig **R**essursforvaltning **A**v **S**kogbestand vha mas**K**inlært, aut**O**matisk kartle**G**ging

TABLE OF CONTENT

ALLSKOG
EID AV SKOGEIERE

ABSTRACT / KORTVERSJON

Prosjektet BRASKOG går ut på å utvikle og teste kvaliteten og nytteverdien av automatisert skogbestandskartlegging med utgangspunkt i fjernmålte datasett.
Her undersøkes spesielt potensiale for "deep learning" metoder av type Convolutional neural networks (CNN). Store datamengder fra tradisjonell skogbruksplanlegging brukes som treningsdata.
I fase 2 av prosjektet ble casestudien fra første BRASKOG prosjektet videreført. I fase 2 ble prediksjonsmodellene ytterlig forbedret. I tillegg til forbedringen ble det fokusert på å lære modellene å predikere en verdi for hvor sikker prediksjonen er. Denne informasjonen ga for treslagsmodellen veldig gode resultater da man kan slå fast om modellen er sikker i sin tolkning eller om manuell tolkning av bestandet fortsatt vil være nødvendig.
Videre ble det jobbet med automatisk bestandsinndeling i fase 2. Det ble både teste en tilnærming der homogene bestand vokser fra minimalt areal til å omfatte et homogent område. Selv om maskinlæringen her viste 'forståelse' for ønsket resultat var overgangen til å få sammenhengende bestand et problem som ikke ble løst i denne fasen av prosjektet. I prosjektet ble det i stedet for fokusert videre på at ALLSKOG skulle etablere modellene i eget miljø. Derfor ble det valgt en enklere tilnærming til bestandsinndeling der den maskinlæret modellen tegner ut tydelige grenser som kan fin inndeles manuell. Dette fører til besparelse på arbeid som ikke trenger spesialkompetanse og er dermed en bra resultatet for prosjektet.

1. INTRODUCTION

A major pillar for the business of ALLSKOG SA is the availability of forest inventory data. The forest inventory data usually contains information on tree species (treslag), maturity class (hogstklasse), forest volume (trevolum), information for biodiversity considerations and conservation demands as well as visualization of forest roads and in some cases expected challenges for logging operations. Forest inventory data is essential for long term sustainable forest management guaranteeing a high level of felling volume throughout the years.
Currently the production line for creating forest inventory data takes advantage of both lidar data and aerial imagery. The lidar data is used to retrieve the tree height and volume, whilst aerial imagery is interpreted by humans to derive the tree species distribution and maturity class among others. The manual interpretation and mapping is very labor intensive work and therefore expensive. Firstly, the forest is divided into stands, which are defined as homogeneous parts of the forest that will be treated as units. Secondly, there are certain attributes for each stand that must be set. The most important attributes are maturity class, siteclass and tree species distribution on which the calculation of volume attributes from lidar data depends. Due to the high costs related to creating the forest inventory, data north of Dovrefjell is lacking, extensive areas have never been inventoried or are outdated.

Within the first BRASKOG project the potential of Deep Learning (DL) methods was evaluated for automating the creation of forest inventories. The main result of the project was several prototype DL models for the classification of forest, tree species and maturity class respectively. These prototypes showed promising results in the study area, but they could not directly be used in an operational setting yet. Additional research and development (R&D) was required to improve the models, especially in terms of robustness and scalability. The main challenge here is to develop DL models that are generic enough such that they can be applied on completely different unseen municipalities. This is very challenging due to spatial variability of the forest and variation in the input imagery. Imagery might be collected at a different time of the year, different time of the day, with different weather conditions, at a slightly different altitude and preprocessing might vary. The appearance of the forest can therefore vary greatly between images. In addition to improving the models the operationalization process requires an effort

In this follow up project the main goal is to automate the creation of forestry inventories with the use of DL methods. Sub goals to achieve this main goal are:

- Continue development of DL models for automated forest, tree species and maturity class classification of aerial imagery, focusing on robustness/scalability.
- Automate the process of stand delineation exploiting the output of the DL models.
- Building an automated processing pipeline for processing large areas deploying models at ALLSKOG

The first two goals are R&D goals, whilst the latter two are operationalization goals.

2. DEEP LEARNING BACKGROUND

Deep learning is a subset of methods within the larger domain of Machine learning. Deep learning methods are based on multilayer artificial neural networks (NNs), hence the name deep. In recent years these techniques have improved the state of the art in speech recognition, computer vision and many other domains *(Lecun et al., 2015)*. Although NNs have been around for several decades, research and applications have exploded in recent years. Mainly because advances in hardware enabled training of more resource demanding complex/deeper networks. The strength of these types of models is that complex non-linear relationships can be mapped/learned from the data, achieving very high accuracy on many kinds of classification tasks. Typically, large datasets are required to learn these relationships and training and running these models is resource demanding. In addition, the increased model size comes with an increased risk of overfitting, proper selection of the train, validation and test sets are therefore of key importance.

Convolution neural networks (CNNs) are a particular type of NN, specifically designed for working on image data. These networks make use of image kernels which are applied on the image, this is called convolution. Major advantages of these image kernels is that weights are shared across the image and connectivity is local. In practice this means the network becomes translation invariant and is simplified as fewer parameters are needed.

CNNs have changed the field of computer vision and are specifically used for image classification, object detection and image segmentation. Image classification is the process of classifying an entire image into a single class. Object detection is the detection of one or more objects belonging to a certain class within the image. Image segmentation is the partitioning of an image into segments with similar characteristics. In essence the image segmentation can be seen as classification on pixel level. By determining the class of every pixel, segments naturally form by connected pixels with the same class. Popular CNN architectures for image segmentation are U-net and segnet *(Ronneberger et al., 2015)(Badrinarayanan et al., 2017)*.

Within the field of remote sensing CNNs have been applied on all types of imagery, from different sensors and platforms. Typical problems for which CNNs are exploited are image classification, object detection and image segmentation *(Zhu et al., 2017)*. Image classification of entire satellite/aerial images is uncommon as these images are typically extremely large, comprising large areas with many different objects/features. Examples do exist and image classification has for example been used for determining whether a satellite image is clouded or not. There are many applications for object detection, examples include detection of vehicles (boats, cars, planes), buildings, roads and trees. Finally, image segmentation is also exploited for many different use cases such as cloud masking and land cover detection. Some applications where CNNs are exploited within the forestry sector specifically are: clearcut detection, forest fire detection, biomass estimation, tree species determination, forest health monitoring, detection of disease outbreaks and drought.

3. STUDY AREA

The study area of the project comprises several municipalities in Trøndelag and Nordland, see figure 1. The selection of these municipalities was based on the availability of recent forest inventory data. The data is split into training, validation and test sets. The training set is used for training the model, the validation set for tuning hyperparameters and the test set for a final evaluation. The hyperparameters are parameters that define the model architecture and control the learning process, such as for example the learning rate or loss function parameters. Since the validation set is used to tweak hyperparameters an independent test set is used for final validation. Within the training municipalities (blue) several tiles are left out for both validation (green) and testing (orange). In addition, several other complete municipalities are involved in the project to test resulting models. For Overhalla the forest management plan was created during the BRASKOG project phase 1, for Rindal and Rana the forest management plan generation is currently nearly finished. The data for these municipalities is used for validation, result description and future planning.

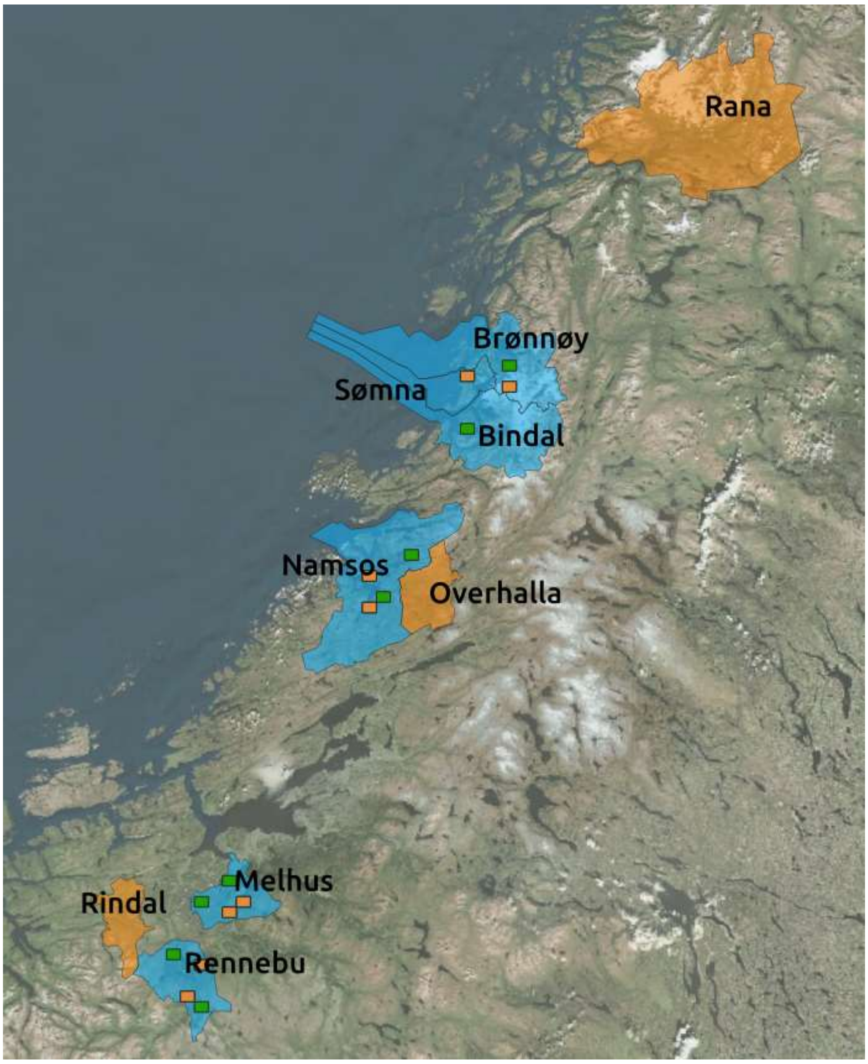

Figure 1: Study area with the municipalities used for training (blue) and testing (orange) and tiles left out for validation (green) and testing (orange).

4. DATASETS

There are several datasets that are exploited during the BRASKOG project. This section shortly describes the main datasets and their role within the project.

4.1. Aerial ortho imagery

The aerial ortho imagery exploited in the project is derived from "Norge i Bilder" *(https://norgeibilder.no/)*. Imagery is downloaded for all the municipalities shown in figure 1. The imagery is coming from different projects and therefore the specifics might vary slightly from project to project. This variation is one of the main reasons to test on several geographically separate locations. An example of an ortho image over a single tile in Melhus is shown in figure 2. The downloaded imagery has 3 bands, RGB and a spatial resolution of 20-25cm. The ortho imagery is generated by mosaicking raw stereo imagery in such a way that the imagery is close to nadir for all locations. The ortho imagery is used as the main input feature for the CNN segmentation models.

Figure 2: Ortho image of a single tile in Melhus.

4.2. Elevation data

The elevation data exploited in the project is derived from “Hoydedata” *(https://hoydedata.no/LaserInnsyn/)*. For several of the municipalities in figure 1 gridded lidar data was downloaded matching the resolution of the aerial imagery. Both Digital terrain model (DTM) and digital surface model (DSM) were retrieved, these layers were used to derive a canopy height model (CHM). The DTM and CHM were evaluated as potential input features for the different segmentation models. An example of these products is shown in figure 3.

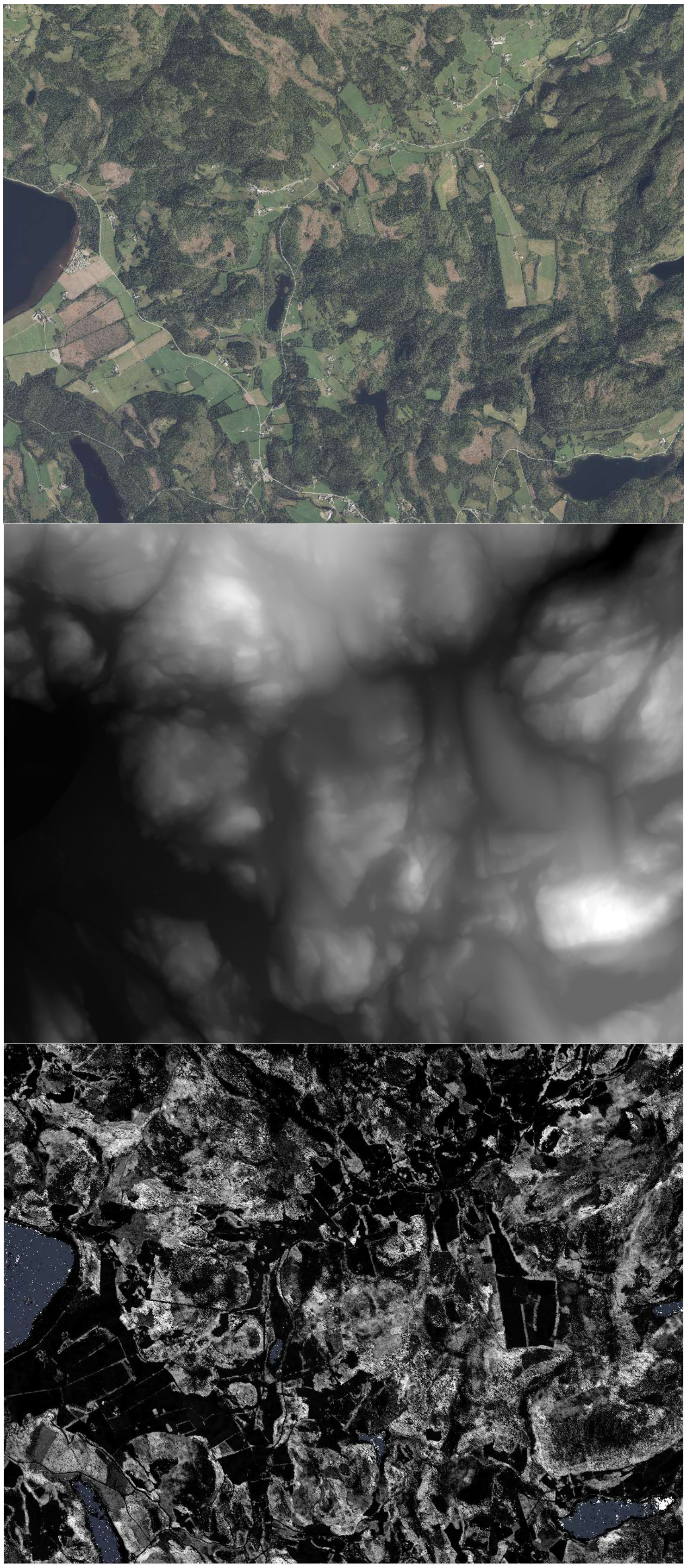

Figure 3: Ortho image (left) DEM (middle) and CHM (right).

4.3. Forest Inventory data

The forest inventory data is the data that is generated by ALLSKOG. Figure 4 shows an example of forest inventory data over a tile in Melhus. The forest inventory consists of forest stands, which are polygons that cover roughly homogeneous areas of forest in terms of tree species, maturity class and density. A typical stand size is around 10.000m2 (10 dekar). Attribute information is attached to these stands giving an inventory of the forest. The most important attributes in the BRASKOG project are the ones related to tree species, maturity class and stand delineation in general. The forest inventory data is used as ground truth data for training and validating of the CNN models.

Figure 4: Forest inventory data over a single tile in Melhus.

4.4. AR5 and AR50

The AR5 dataset is the Norwegian national land usage dataset aiming to cover detailed scales of around 1:5000 (*NIBIO ar5*). The AR50 is the national Norwegian land usage dataset covering the entire country (NIBIO AR50). The AR50 data is a generalization of AR5 data completed with satellite based information in mountainous areas (see figure 5). Like the forest inventory data the AR5 data is used as training and validation data. The forest inventory data and AR5 used in this project only covers the production forest for ordered forestry management plans. AR50 is freely available and used to cover the images as a complete dataset.

Figure 5: AR50 data for non-forested areas over a single tile in Melhus.

5. CNN SEGMENTATION MODELS

5.1. Introduction

The main underlying technology for automating the creation of the forest inventory are different CNN segmentation models with a U-net architecture. Within the project models are developed for tree species classification, maturity class classification and forest/non forest classification. The main input for these segmentation models are the aerial ortho imagery. But additional input features such as the site class, digital elevation model (DEM) and canopy height model (CHM) have been evaluated throughout the project. The segmentation models yield a prediction on pixel level. However, a larger window within the image is exploited to make this prediction. The used window size is 256x256 pixels corresponding to roughly 50x50m since the imagery is at a resolution of 20cm.
The prediction for each pixel is a distribution over the different classes. Meaning that a value between 0-1 is given for each class and values always sum up to 1. The higher the value the more certain the model is about the prediction. However, these values should not be directly interpreted as probabilities as they are not calibrated. Calibration of these probability is not possible with the available data, since accurate reference data is only available at stand level.

5.2. Validation methodology

In order to validate the tree species and maturity class segmentation models these pixel values should be aggregated to stand level, because the ground truth data is only available for the stands and not individual pixels. If the tree species distribution in a stand is 50% spruce and 50% pine it does not mean that this is true for the individual pixels within this stand. A pixel can be completely spruce or pine or a small open space or be the boundary of one of these classes. Likewise, the maturity class tells something about the average maturity of the trees in the stand, but it doesn't mean that all trees are equally mature.

There are different ways in which the aggregation can be done. In the ideal situation each pixel would belong to a specific class. For aggregation the number of pixels of each class within the stand could then just be counted giving the distribution of tree species or most common maturity class within each stand. The difficulty comes from the fact that model outputs on pixel level are distributions already. In order to aggregate on stand level the average of all pixel distributions is taken. This distribution is compared to the reference or ground truth distribution in different ways to evaluate model performance. The argmax of the distributions is taken to derive metrics like the confusion matrix and f1 score.

The tree species and maturity class results presented in this chapter are derived by running the models on 1000 randomly selected stands in Overhalla. The entire municipality of Overhalla, Rindal and Rana were left out of the training data so that it can be used for independent testing. This is in line with one of the main goals, to make robust models that function across different municipalities.

5.3. Tree species

An example of a complete tree species prediction for an entire tile can be seen in figure 6. The model outputs 4 classes: spruce, pine, deciduous and background (everything that is not forest). This particular tile shows a typical situation where there is some birch forest present down in the valley close to agricultural areas. The spruce and pine forest is more abundant covering most part of this tile.

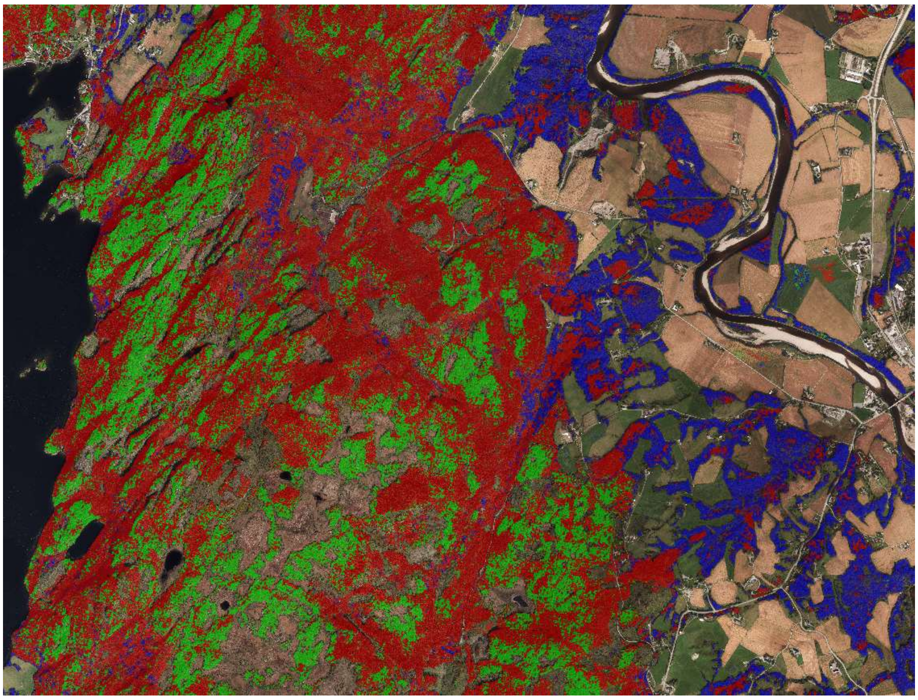

Figure 6: Tree species, spruce (red), pine (green) and deciduous (blue) prediction raster for a single tile in Melhus.

The confusion matrix below (figure 7) shows the performance across the different tree species classes. Note that as explained in the previous section these values are derived by comparing the prediction with the manual labels at stand level. The confusion matrix only indicates whether the dominant tree species is classified correctly and in case it is not which other tree species was predicted. From this matrix we can see that the model is slightly biased. Spruce stands are more frequently classified as pine than the other way around. Similarly deciduous stands are too often classified as spruce.

		pred		
		spruce	pine	deciduous
true	spruce	75.5%	21.6%	2.9%
	pine	5.8%	93.3%	1.0%
	deciduous	40.2%	9.2%	50.6%

Figure 7: Confusion matrix for the tree species (treslag) model based on 1000 random stands in Overhalla.

Figure 8 gives a more detailed view, taking the distribution into account and not only the dominant tree species. The aim of this table is **to reliably find stands that are dominated by a specific tree species**. These stands should in practice be easier compared to stands that are a mix of species. The way the table works is that every row represents a subset of the stand data, based on some specific filtering on the prediction values. For example the first row are all the stands for which the model predicts spruce >60, which are in total 305 stands. The columns show the true spruce percentage for these stands. A higher threshold can thus be taken to be more certain that the filtered stands are indeed spruce stands, but fewer stands will be filtered out. There is thus an important tradeoff between the amount of stands for which the attributes can be derived automatically and the certainty of the result. In this case it could for example be decided that for spruce a threshold of 70 is used for pine 70 and for deciduous 80.

	0-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80-90	90-100	total pred	total true	%
spruce > 60	1	1	2	6	4	12	15	33	104	127	305	599	50.9
spruce > 70	0	0	0	1	0	6	4	14	55	94	174	535	32.5
spruce > 80	0	0	0	0	0	0	0	3	15	38	56	415	13.5
pine > 60	11	9	6	5	7	4	15	20	22	11	117	88	133
pine > 70	1	0	1	1	1	0	9	10	11	10	45	67	67.2
pine > 80	0	0	0	0	0	0	0	0	0	1	1	39	2.6
deciduous > 60	2	1	1	0	1	5	9	5	5	4	33	60	55
deciduous > 70	0	1	1	0	0	2	7	2	4	4	21	30	70
deciduous > 80	0	0	0	0	0	1	2	1	2	1	7	14	50

Figure 8: Potential prediction thresholds for the various tree species in order to automatically derive attribute information.

5.4. Maturity class

An example of a complete maturity class prediction for an entire tile can be seen in figure 9. The model outputs 5 classes: HK2 (ungforest), HK3, HK4, HK5 (mature forest) and background (everything that is not forest). HK stands for hogstklasse, meaning maturity class. HK5 prediction is most abundant, but patches of HK4, HK3, HK2 can be seen throughout the forest. In some cases sharp boundaries are present, showing the potential of finding stand boundaries using this raster.

Figure 9: Maturity class, 2 (lightest green) - 5 (darkest green) prediction raster for a single tile in Melhus.

The confusion matrix below (figure 10) shows the performance across the different maturity classes. From this matrix we can see that the results for HK2, HK3 and HK5 are reasonable and that the main issue is with HK4 predictions. HK4 stands are often predicted as either HK3 or HK5. Obviously in the case of maturity class neighboring classes appear more similar and therefore it is expected that most errors are cases where the prediction is one maturity class off.

		pred			
		hk2	**hk3**	**hk4**	**hk5**
true	**hk2**	62.5%	32.1%	0.9%	4.5%
	hk3	13.6%	71.1%	7.9%	7.4%
	hk4	5.3%	27.2%	23.9%	43.6%
	hk5	2.4%	3.2%	8.1%	86.2%

Figure 10: Confusion matrix for the maturity class (hogstklasse) model based on 1000 random stands in Overhalla.

Figure 11 gives a more detailed view, taking the distribution into account and not only the dominant maturity class. The aim of this table is to reliably find stands of a certain maturity class. The way the table works is that every row represents a subset of the stand data, based on some specific filtering on the prediction values. For example the first row are all the stands for which the model predicts HK2 >50, which are in total 73 stands. The columns show the true maturity class of these stands. Like for the tree species there is a tradeoff between the amount and the certainty of the results. In this case it could for example be decided to use a threshold of 60 for HK2, 50 for HK3 and 50 for HK5, whilst filtering on HK4 is omitted.

	true				
	hk2	**hk3**	**hk4**	**hk5**	**% of total**
hk2 > 50	46	20	4	3	65.18%
hk2 > 60	34	13	1	1	43.75%
hk2 > 70	19	10	0	1	26.79%
hk3 > 50	13	117	16	1	37.60%
hk3 > 60	1	18	1	0	5.12%
hk3 > 70	0	0	0	0	0.00%
hk4 > 50	0	0	4	0	1.65%
hk4 > 60	0	0	0	0	0.00%
hk4 > 70	0	0	0	0	0.00%
hk5 > 50	0	4	67	166	95.95%
hk5 > 60	0	2	36	88	51.01%
hk5 > 70	0	1	13	23	14.98%
total	112	391	243	247	

Figure 11: Potential prediction thresholds for the various maturity classes in order to automatically derive attribute information.

5.5. Forest

An example of a complete forest prediction for an entire tile can be seen in figure 12. The model output is binary: forest and background (everything that is not forest). The relatively sharp boundaries between forest and no forest show the potential of finding the outer edge of the forest. For the forest model we are not going in further detail with confusion matrix and additional tables since the forest model is not used to fill attribute information.

Figure 12: Forest prediction raster for a single tile in Melhus.

6. FOREST INVENTORY GENERATION

The goal of the project is to automate the creation of forestry inventories with the use of the developed segmentation models. The output rasters of the segmentation models can be used to automate the generation of the forest inventory in two main ways. First off all the prediction rasters can be used to generate attribute information for an existing layer of forest stands. Secondly the prediction rasters can be used to support delineation of the forest stands themselves. In this section both the generation of attribute information and delineation of stands are discussed.

6.1. Attribute generation

As described in section above, the output of the segmentation model is a raster with a prediction for each pixel. In order to derive the attribute information for forest stands the pixel values are aggregated to stand level. Like for the validation the aggregation is done by taking the mean of the pixel distributions. This results in a distribution on stand level. In the maturity class case we are only interested in a single maturity class per stand. Therefore simply the maturity class that has the largest share of the distribution is selected. How dominant the largest share is gives an indication about the certainty.

6.2. Stand delineation

The definition of a forest stand is a homogeneous patch of forest in terms of tree species, maturity class and sparsity of the forest. The output of the segmentation models are rasters for forest, tree species and maturity class. These could thus potentially directly be used to delineate forest stands. Within the project two main approaches were evaluated.

The first approach is a region growing approach where the aim is to derive forest stands directly from the classification rasters. This approach proved to be very challenging as most boundaries are gradual transitions rather than sudden boundaries, resulting in relatively arbitrary stands.
The second approach is to find the most prominent boundaries rather than the stands themselves. The main idea here is that the “easy” boundaries can be automatically derived. These boundaries can serve as a starting point for a forest expert who manually finishes the delineation by finding the “difficult” boundaries. The output of the forest model can be used to find transitions between forest and no-forest. Similarly the output of the tree species and maturity class models can be used to find sudden transitions within tree species and maturity class. For the binary forest rasters boundaries are derived by thresholding the raster at 0.5. The maturity class and tree species have multiple classes so the equivalent would be thresholding on the most dominant class. However, for tree species the forest might be a mix of species and we want to find sudden changes in the composition of the forest rather than a tipping point in the dominant tree species. Therefore our approach is to search for large gradients in the tree species output raster to find these sudden changes. In addition, different pre- and post- processing steps like gaussian smoothing, filtering of holes and islands are applied to improve the derived boundaries (see figure 13)

Figure 13: Forest delineation created in operations project at ALLSKOG

7. OPERATIONALIZATION

7.1. Dagger

One of the main challenges in terms of operationalization is the efficient processing of large areas. In order to achieve this goal the in-house developed processing framework dagger is used. Among others, the Dagger framework allows for task scheduling, resource allocation, coupling of inputs/outputs, re-usability of previously computed products and more advantages which allow for the generation of an efficient and robust processing chain.

7.2. Processing graphs

Figure 14 shows the dagger graph that is used to run segmentation models on large areas and aggregate results on stand level to generate attribute information. In the configuration file the area of interest (AOI), CNN model, ortho imagery and forest stand data is specified. The AOI can be specified in the form of a municipality name or a specific tile id. The actual geometry of the AOI is queried from the database and used to determine which tiles should be processed. Tiling is necessary to avoid exploding file sizes and ensure scalability. For each tile a separate processor is triggered that applies the trained segmentation model on the ortho imagery to generate a prediction raster. The model can be a tree species, maturity class or other model and is specified in the config. All the resulting prediction rasters are input for the zonalmean processor. Here the forest stand polygons within the AOI are derived from the database and predictions are aggregated on stand level. The result is a forest stand polygon layer with attributes containing the predicted values by the CNN segmentation model.

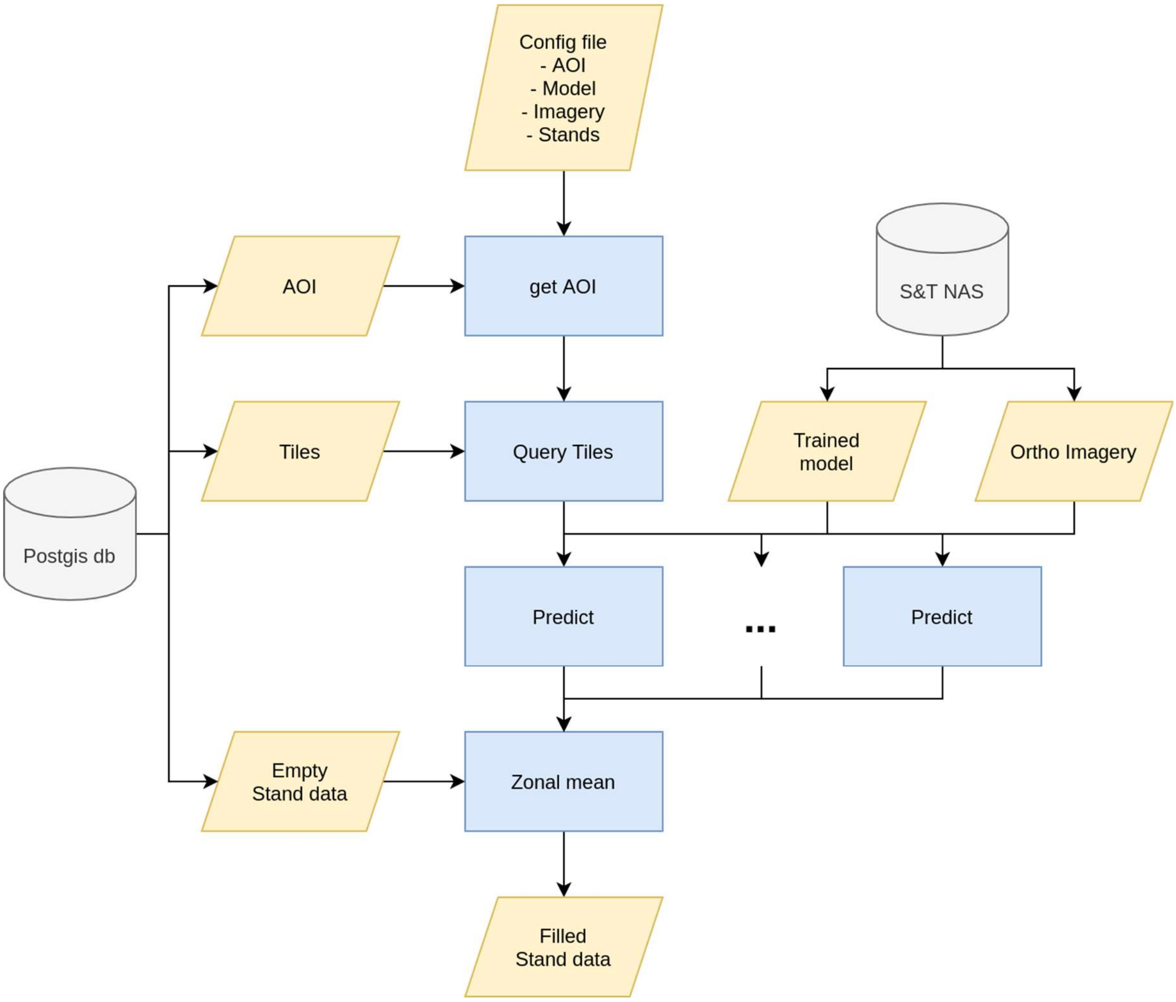

Figure 14: Dagger graph that is used to run segmentation models

7.3. Deploying models/processors

Although dagger has been used for internal processing at S&T, it has not been deployed to ALLSKOG's system. The deployment of dagger, or more generally deployment of any software stack still in development, comes with a set of challenges which are outside the scope of this project. Instead of deploying dagger as a whole we decided to go for a simpler solution: wrapping the processing tools inside a single docker image. This image then ships with all required dependencies and ensures ease-of-use on ALLSKOG's system.

8. DELIVERIES

The project has resulted in a number of deliveries from S&T to ALLSKOG which are evaluated in an operational setting. The deliveries can be subdivided into 1) data deliveries where processing is done by S&T and 2) software deliveries where a tool or software component is delivered.

8.1. Data deliveries

For the municipalities Rindal and Rana tree species and maturity attributes were generated, using the processing graph presented in figure 14. Stand data in Rana and Rindal attribute information was derived manually by the end of the project; these stands are used for validation purposes. In chapter 5 performance results were shown for Overhalla, which gives an indication of the generalization power of the models. Meaning how well they perform on imagery of a municipality that was not part of the training data. However, the trained models are expected to perform differently across municipalities, therefore this additional data in Rindal and Rana is useful to get a more accurate indication of the performance in these areas.

8.2. Software deliveries

For the deployment of models at ALLSKOG, forest delineation was selected as a test case. Forest delineation is part of the stand delineation and is the first step in finding stand boundaries in an automated way. The forest delineation delivered as a module to run at ALLSKOG focuses solely on boundaries between forest and no-forest and not on internal stand boundaries within the forest. The reason the forest delineation was selected is that the forest model was deemed most robust order to make the process tested in ALLSKOG environment. The delivery itself consists of an inference-, vectorization- and post processing module. The inference module contains the trained forest model, it allows the model to be run on images of arbitrary sizes. The vectorization module thresholds the output forest raster to derive forest polygons. During post processing small islands and holes are removed. The forest delineation is containerized as a single application using docker ensuring it can be shipped and run reliably at ALLSKOG.

9. CONCLUSIONS AND FUTURE WORK

ALLSKOG validated the deliveries of all model results with usual methods for forest inventory, using mostly manual remote sensing in 2D or 3D.
For the main tree species model the conclusions in this report are done by validating the forest inventory attribute setting and comparing based on stand number and area for the forest inventory done in the municipalities. For the stand delineation the delivered model was used to create stand delineation data which was implemented in the usual process of the forest inventory as a foundation to build up on.
For Rindal the dominant tree species was detected correctly for close to 30% of the stands in the municipality (figure 15).
It is however not the amount of number/area of stands that is the most valuable part of the correct predictions. The most important statement is that the confusion matrix for the model could be used to find the thresholds to decide which model is certain about the predictions. Having the combination of the model being certain and the correct labeled stands is the main value of the evaluation.

Dominerende treslag	Antall	Areal
Prod skog taksert Rindal kommune	4164	39 847
Antall/Areal sikre bestand i Rindal for bontre	1218	11 243
Prosent av antall/areal	28,6	28,2

Figure 15: Rindal stands validated as correct from BRASKOG tree species model

For Rindal the thresholds for spruce were set at a certainty lever of >60, while the threshold was > 80 for the predicted stand information of BRASKOG models.(figure 16)

		SANN									
		(0.0, 10.0]	(10.0, 20.0]	(20.0, 30.0]	(30.0, 40.0]	(40.0, 50.0]	(50.0, 60.0]	(60.0, 70.0]	(70.0, 80.0]	(80.0, 90.0]	(90.0, 100.0]
Pred	volumpros_g > 80	0,00 %	0,00 %	0,00 %	0,00 %	0,00 %	0,00 %	0,41 %	0,83 %	13,28 %	85,48 %
	volumpros_g > 70	0,00 %	0,00 %	0,23 %	0,00 %	0,23 %	0,00 %	0,70 %	2,09 %	22,97 %	73,78 %
	volumpros_g > 60	0,17 %	0,00 %	0,33 %	0,00 %	0,33 %	0,33 %	1,34 %	4,17 %	26,71 %	66,61 %
	volumpros_f > 80	1,08 %	1,08 %	1,08 %	2,15 %	0,00 %	0,00 %	2,15 %	10,75 %	37,63 %	44,09 %
	volumpros_f > 70	2,97 %	1,49 %	2,23 %	1,86 %	0,37 %	2,60 %	7,43 %	14,13 %	39,03 %	27,88 %
	volumpros_f > 60	5,57 %	2,90 %	5,12 %	4,01 %	2,23 %	2,90 %	8,02 %	13,36 %	33,18 %	22,72 %
	volumpros_l > 80	0,00 %	0,00 %	0,00 %	0,00 %	0,00 %	14,29 %	0,00 %	0,00 %	0,00 %	85,71 %
	volumpros_l > 70	0,00 %	0,00 %	0,00 %	4,17 %	0,00 %	4,17 %	4,17 %	8,33 %	12,50 %	66,67 %
	volumpros_l > 60	0,00 %	0,00 %	0,00 %	2,04 %	0,00 %	2,04 %	6,12 %	6,12 %	22,45 %	61,22 %

Figure 16: Confusion matrix for Rindal with certainty thresholds in percent

The maturity class model was evaluated in current forest inventory projects as well. Even the correct maturity class was promising for the control results for Overhalla in chapter 5, unfortunately the results of the confusion matrix and thereby the model's certainty of it's own prediction was not giving good enough results. In this case even if the results are correct, the manual process still would have to determine the correctness which leads to a minimum in production savings (figure 17).

	true				
	hk2	hk3	hk4	hk5	% of total
hk2 > 60	0	0	0	0	0.00%
hk2 > 70	0	0	0	0	0.00%
hk2 > 80	0	0	0	0	0.00%
hk3 > 60	0	2	0	0	0.30%
hk3 > 70	0	1	0	0	0.15%
hk3 > 80	0	0	0	0	0.00%
hk4 > 60	0	0	0	0	0.00%
hk4 > 70	0	0	0	0	0.00%
hk4 > 80	0	0	0	0	0.00%
hk5 > 60	2	71	257	600	100.43%
hk5 > 70	0	15	120	367	54.21%
hk5 > 80	0	2	24	130	16.85%
total	120	661	559	926	

Figure 17: Potential prediction thresholds for the various maturity classes Rindal

Both the tree species and maturity class models can to some extent be used to generate stand attribute information in an automated way. There is however a tradeoff between the amount of stands for which tree species and maturity class can be derived and the expected quality of the result. For tree species models the results are able to be used in production prefilling stands with values for a given certainty depending on each project. It is not possible to decide in this state a in what scope the model will be used to fill stands in upcoming projects but it definitely can be used. For maturity class the model needs to be improved further.

Additional improvement to these classifiers would mean that reliable results could be retrieved for a larger share of the stands. Two main challenges we identified are the presence of noisy labels and the large amount of variation across imagery and different municipalities. Future work could therefore focus on improving the current setup with semi- and weakly supervised approaches. With semi-supervised approaches unlabeled imagery can be used in addition to make models more robust to variations. Weakly supervised approaches intend to handle noisy training data.
For maturity class model a further main thought is to make site class as in input data. This is due to new research on determining site class automatically. This would make the site class parameter in input and knowing that maturity class, site class and age are a combined description of the forest inventory attributes the site class as input parameter is expected to make a significant difference in the predictions and certainty level.

In the just recently started forestry inventory project for Oppdal ALLSKOG run the stand delineation model on the imagery mainly covering the AOI. In this case we chose to run the model on the newest available orthophoto project and only on this not combining different imagery on the same run.

ALLSKOG
EID AV SKOGEIERE

The outcome presented well differentiated forest/no forest stand delineation helped the manual work to use time on the obvious borders (Fig 18). Unfortunately, we discovered not only few expected (Fig 19) and not expected border effects at the all edges of each tile (Fig 20).

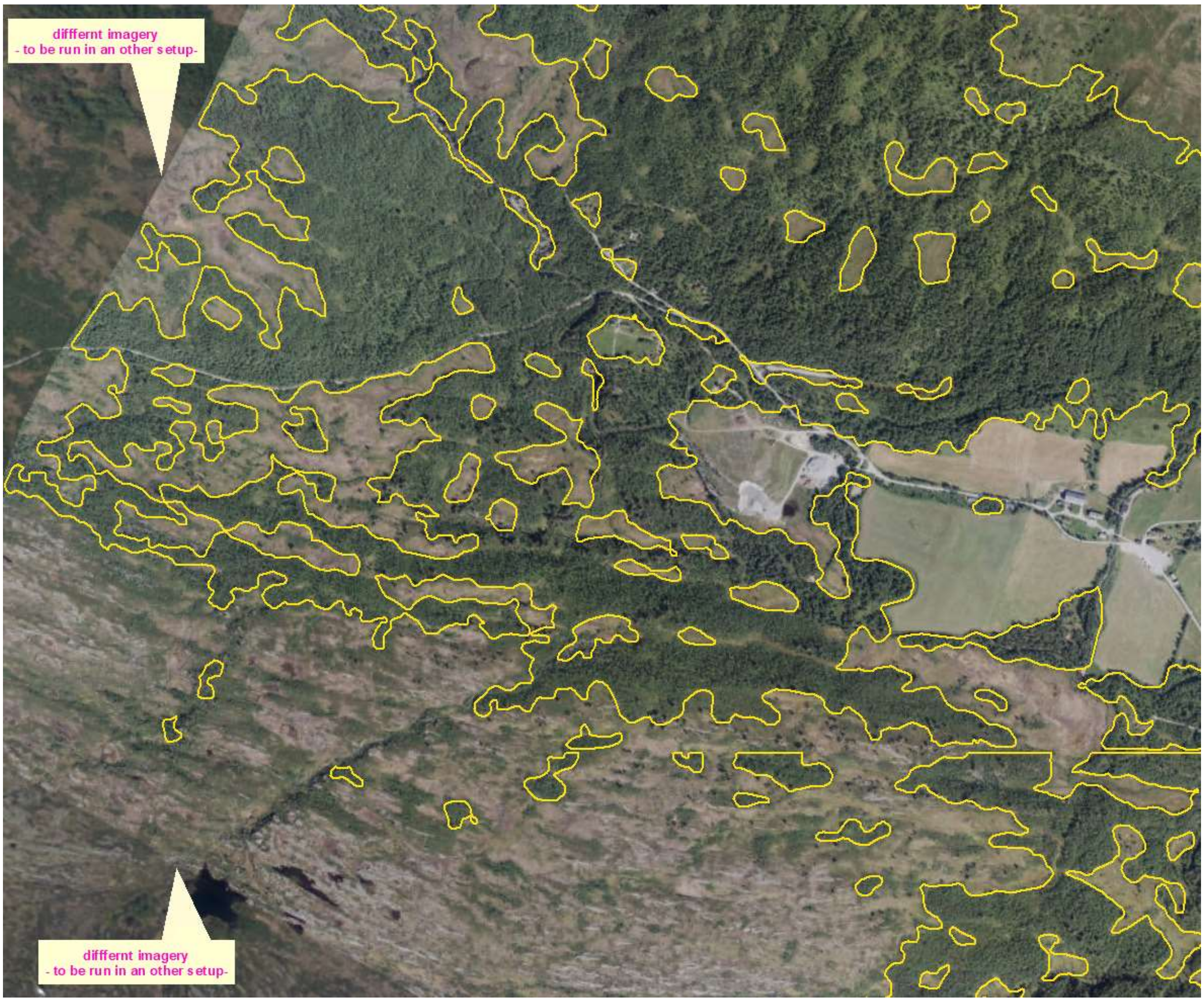

Figure 18: Forest delineation created in operations project at ALLSKOG

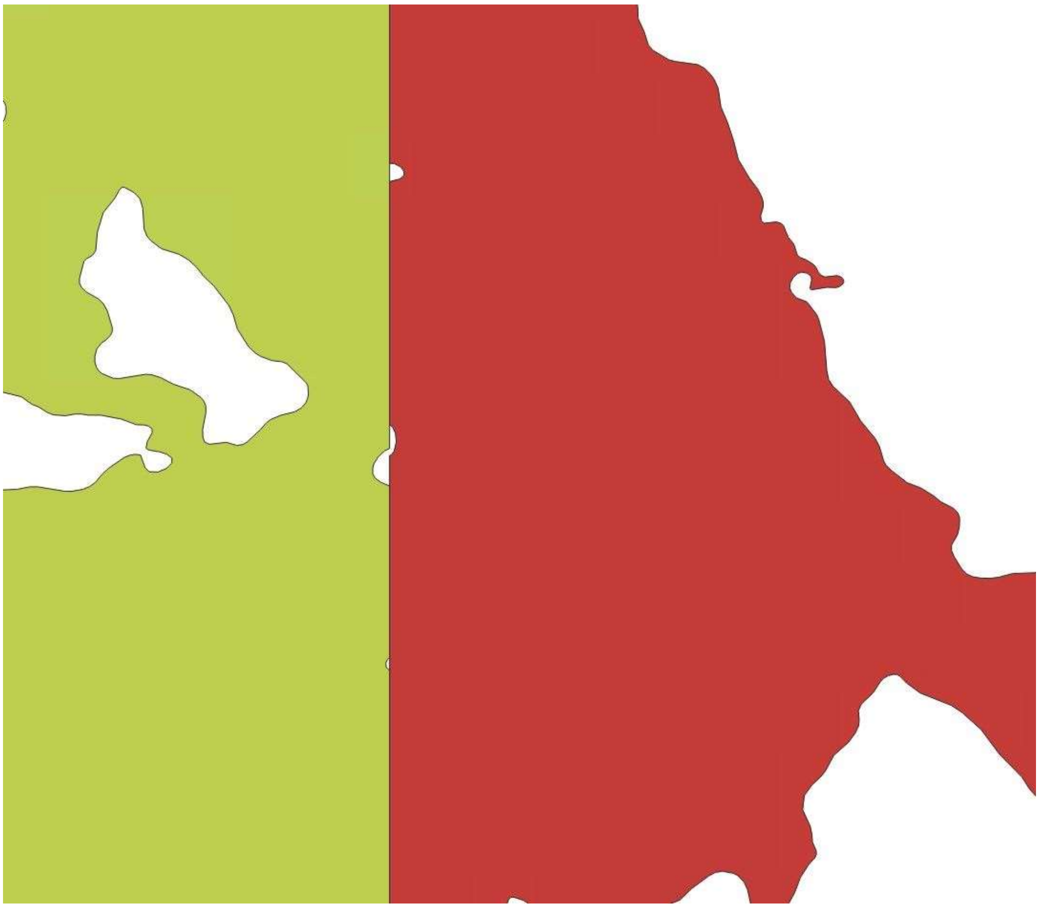

Figure 19: Expected edge effects for the delination model - smal area are not fitting

Figure 20: Gap between tiles - not expected edge effect of the model

The expected edge problems like in figure 19 were on a minimal level and easy to fix manually with the normal stand delineation process.
The unexpected edge problem was within this production time consuming to fix. In this case as the forest needs to start to work the fix was done manually. BRASKOG project did come up with possible solutions to fix the unexpected edge problems. Both running virtual raster or a differently organized running of the model are approaches to look into when continuing the development of the model.

Internal boundaries within the forest are more challenging. The tree species and maturity class rasters show the potential of finding the boundaries to some extent. However, constraints on the shape of the stands and the typically gradual transitions within the forest make this a very challenging problem. Future work could focus on including constraints in the model optimization. For example, penalizing transitions in maturity class and tree species to enforce smoother output rasters.

In terms of operationalization, the current solution to deploy models at ALLSKOG is to containerize them using docker. This works well for deploying individual processors or small processing chains that can be wrapped in a single container. However, this approach is not ideal for complex batch processing jobs or long running production jobs. Such jobs are typically implemented in dagger by S&T, for which deployment comes with a number of challenges and requirements to system administration on the deployed infrastructure. In the future a service based solution for complex processing jobs should be considered.

10. REFERENCES

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444

Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.

Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). Segnet: A deep convolutional encoderdecoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(12), 2481-2495

Zhu, X. X., Tuia, D., Mou, L., Xia, G. S., Zhang, L., Xu, F., & Fraundorfer, F. (2017). Deep learning in remote sensing: A comprehensive review and list of resources. IEEE Geoscience and Remote Sensing Magazine, 5(4), 8-36