

# DETEKTERING AV BYGGGNADER MED MASKININLÄRNING

---

ERIK NILSSON 2024-04-17

LANTMÄTERIET



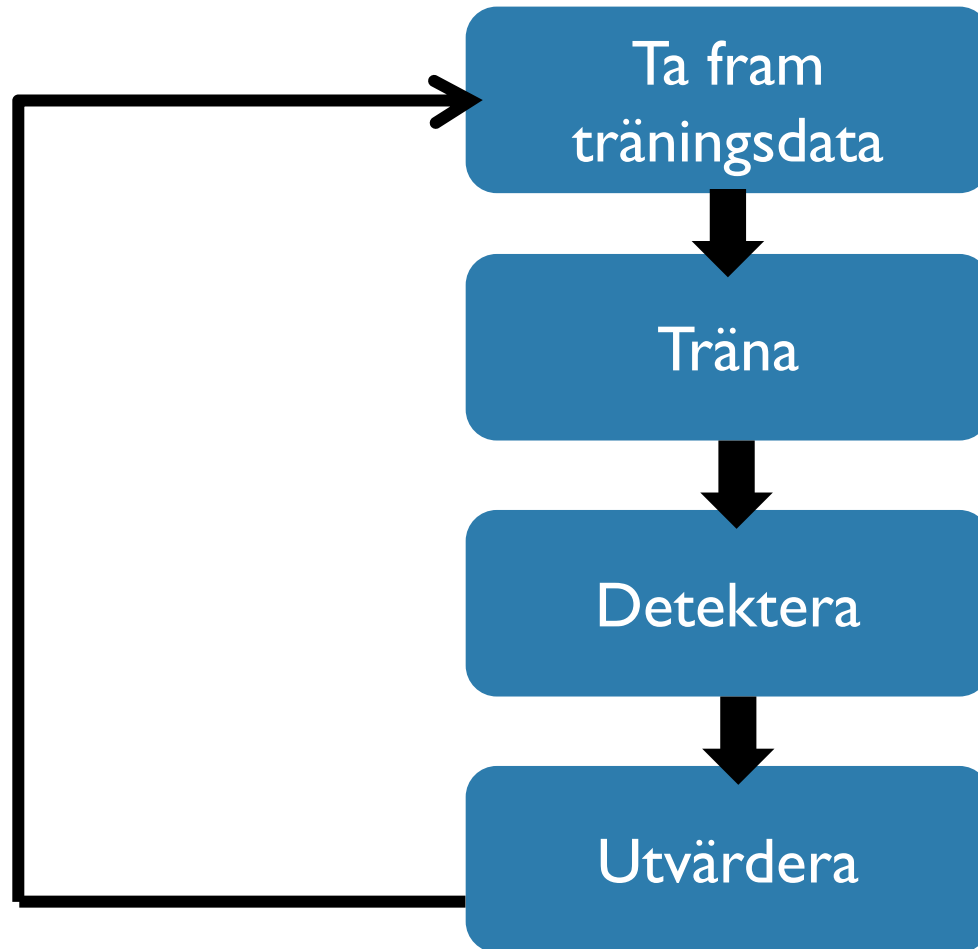
# MASKININLÄRNING INOM GEOGRAFISK INFORMATION

- Vi har en grupp på 6 personer (ca 2 heltider totalt) som jobbar med utveckling av maskininläring inom geografisk information.
- Förutom byggnader har vi även börjat titta på detektering av vägar (främst skogsvägar).
- En kandidat att titta vidare på är klassificering av laserdata.
- Utöver det så har vi och har haft några examensarbeten inom området.

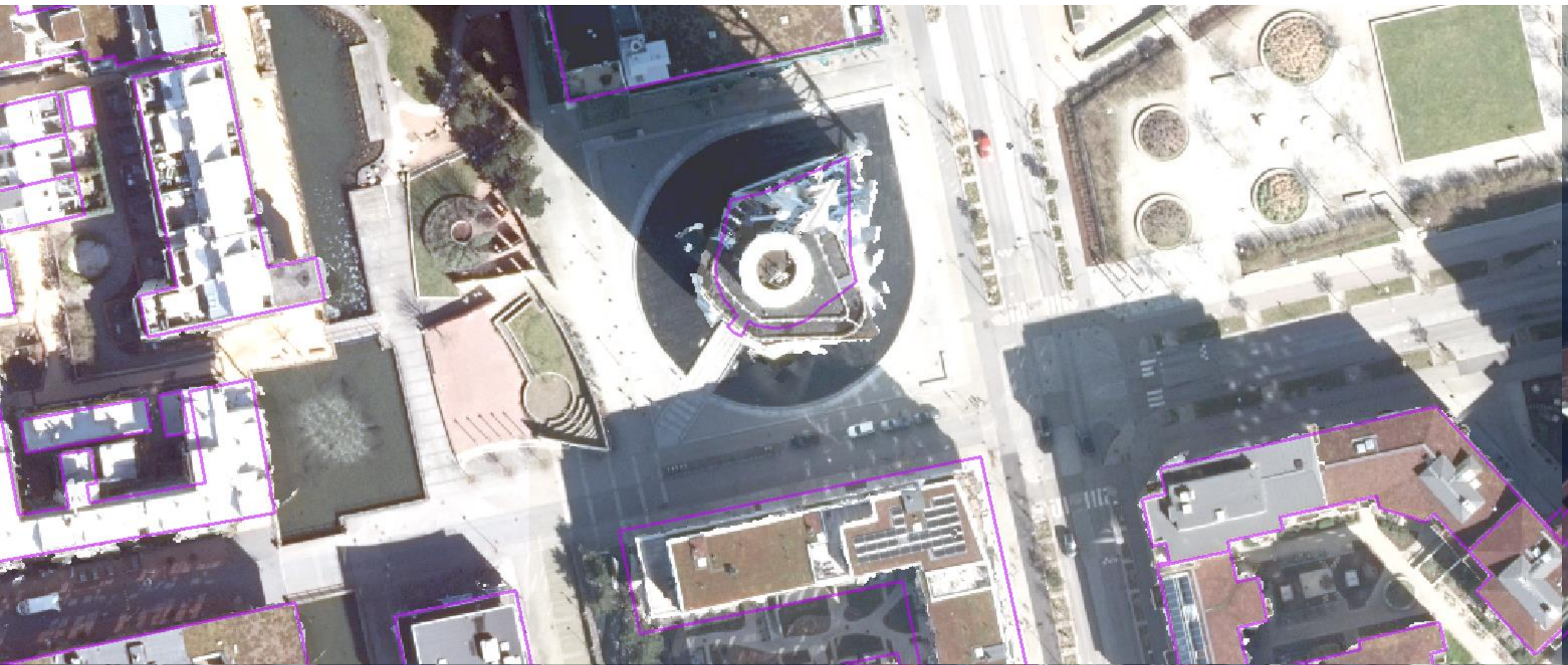
# VAD ÄR MASKININLÄRNING (ML)?

- Kortversionen: ML är ett delområde av Artificiell Intelligens som handlar om metoder för att träna datorer att upptäcka och lära sig regler för att lösa en uppgift.
- I vårt fall betyder det att vi tränar en modell med det indata som vi vill använda samt ett "facit" på vad vi vill att modellen ska hitta.

# UTVECKLA EN MASKININLÄRNINGSMODELL



# ORTOFOTO VS SANT ORTOFOTO



# HÖJDMODELLER

Ythöjdmodell (DSM)

Markhöjdmodell (DTM)

båda är ”höjd över havet”

Normaliserad ytmodell=

$nDSM = DSM - DTM$

”höjd över mark”

# TRÄNINGSDATA

Träningsdatat måste ha:

- **Hög kvalitet**, dvs korrekta geometrier och byggnaderna måste finnas både på ortofoto/höjdmodell och i facitdata.
- **Diversitet**, dvs många olika typer av byggnader/terrängtyper/fotoförhållanden.
- **Kvantitet**, dvs många byggnader att träna på.

# TRÄNINGSDATA

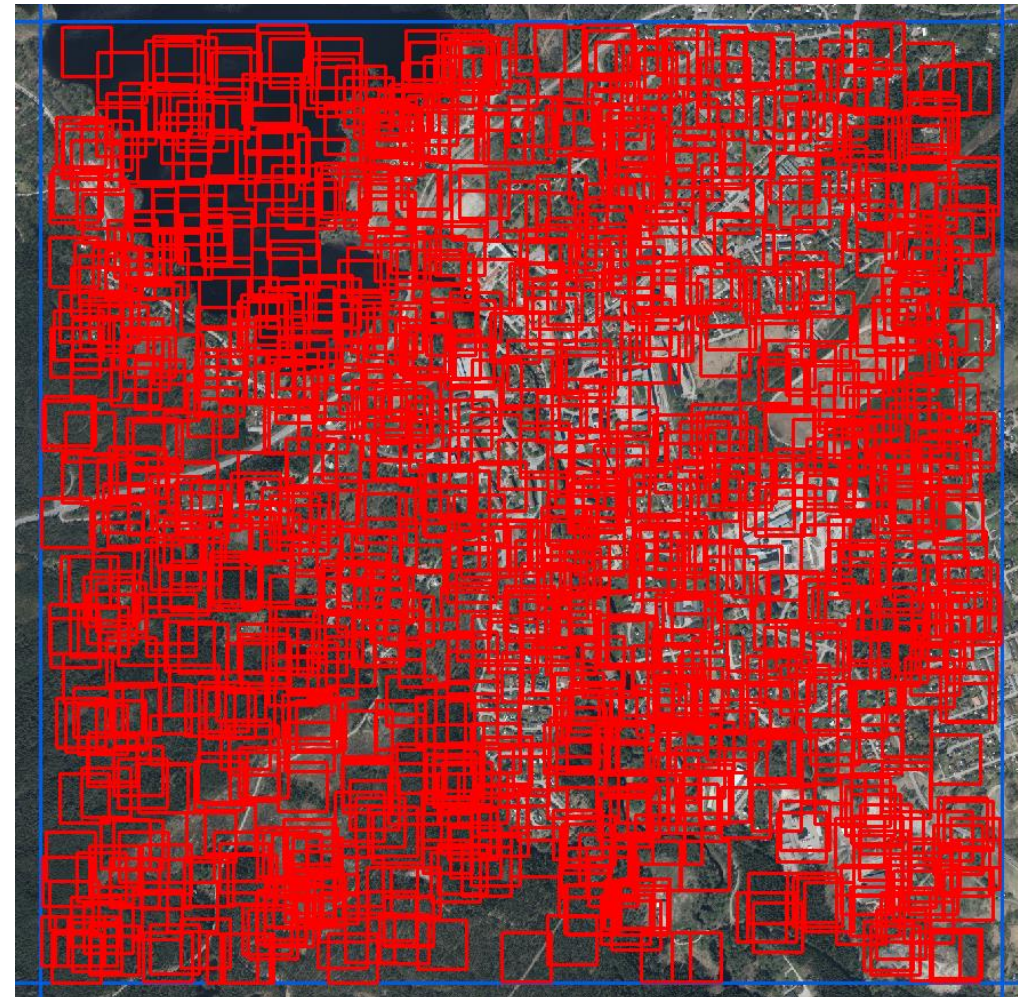
- Det byggnadsdata som finns i Lantmäteriets databaser är generellt sett för dåligt att använda som det är för ML-träning rakt av.
  - Vissa kommuner levererar ofta, korrekt och detaljerat data.
  - Andra kommuner har inte särskilt god aktualitet
  - Lantmäteriet lägger bara in byggnader eller ändringar större än 15-20 m<sup>2</sup> vid flygbildstolkning.
- Vi har slumpat fram rutor med data. Dessa rutor har manuellt kontrollerats och bedömts som Godkänd/Icke Godkänd.
- Rutor har skapats på specifika ställen där det behövs för att träna/träna bort en viss företeelse tex landningsbanor, stora byggnader, berghällar, broar etc.
- Vi har också karterat två mindre områden (3\*3 km) väldigt noggrant och använt detta för att generera träningsdata.



# TRÄNINGSDATA PÅ OLIKA SÄTT



Så här har vi jobbat:  
Inom ett område på 2,5x2,5 km kan vi kanske ha 5 godkända rutor.  
Med lite rotationer och spegelvändning av bilder kan vi få ca 10-20 tiles.



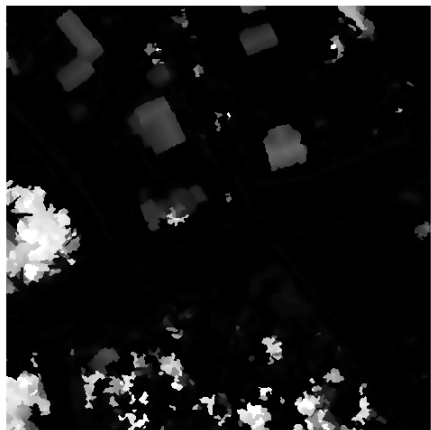
Vi har testat att kartera sammanhängande områden med korrekta byggnader. Om ett område har helt korrekta byggnader kan man teoretiskt generera flera tusen rutor att träna på.

# TRÄNING

- Vi använder "rutor" som är 512\*512 pixlar stora. Med det data vi använder som har upplösningen 0,25m/pixel så innebär det 128\*128m på marken.
- Vi genererar tif-filer av allt träningsdata. Det gör att vi kan använda dessa för träning i framtiden även om datat skulle ändras i verkligheten eller om det kommer nya bilder.
- Idag har vi ca 18000 godkända rutor.
- Basen i all träning vi gjort har varit "True orthos" med R, G, B och IR- kanaler samt rasteriserade byggnader.
- Utöver det har vi testat med ythöjdmodell (DSM), ythöjdmodell+markhöjdmodell (DSM och DTM) samt **normaliserad ythöjdmodell (nDSM)**. (Alla ythöjdmodeller framställda från flygbilder)
- Träningen av en ML-modell är väldigt beräkningsintensiv - kraftfull GPU (grafikkort) krävs
- En träning kan fortgå i allt ifrån någon timme till flera veckor. I vårt fall 4-5 dygn
- Att träningen är så pass tidskrävande skapar en del problem. För att kunna spåra hur olika förändringar i inställningar eller träningsdata mm slår igenom så bör man inte ändra flera saker samtidigt. Detta gör att det tar en hel del tid att utvärdera hur olika ändringar påverkar slutresultatet.



+



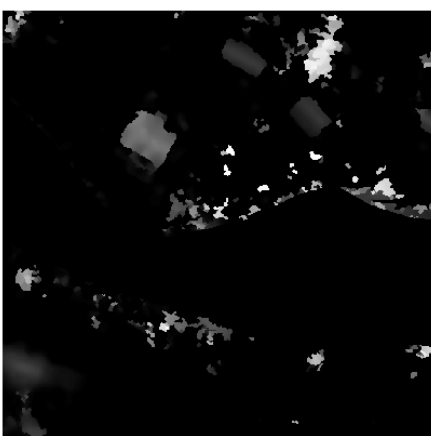
=



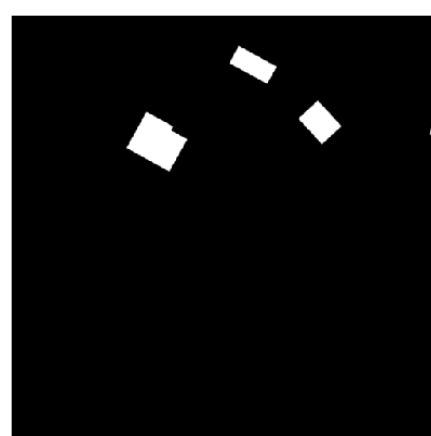
Träning



+



=



+






=


?


Detektering


# DETEKTERA

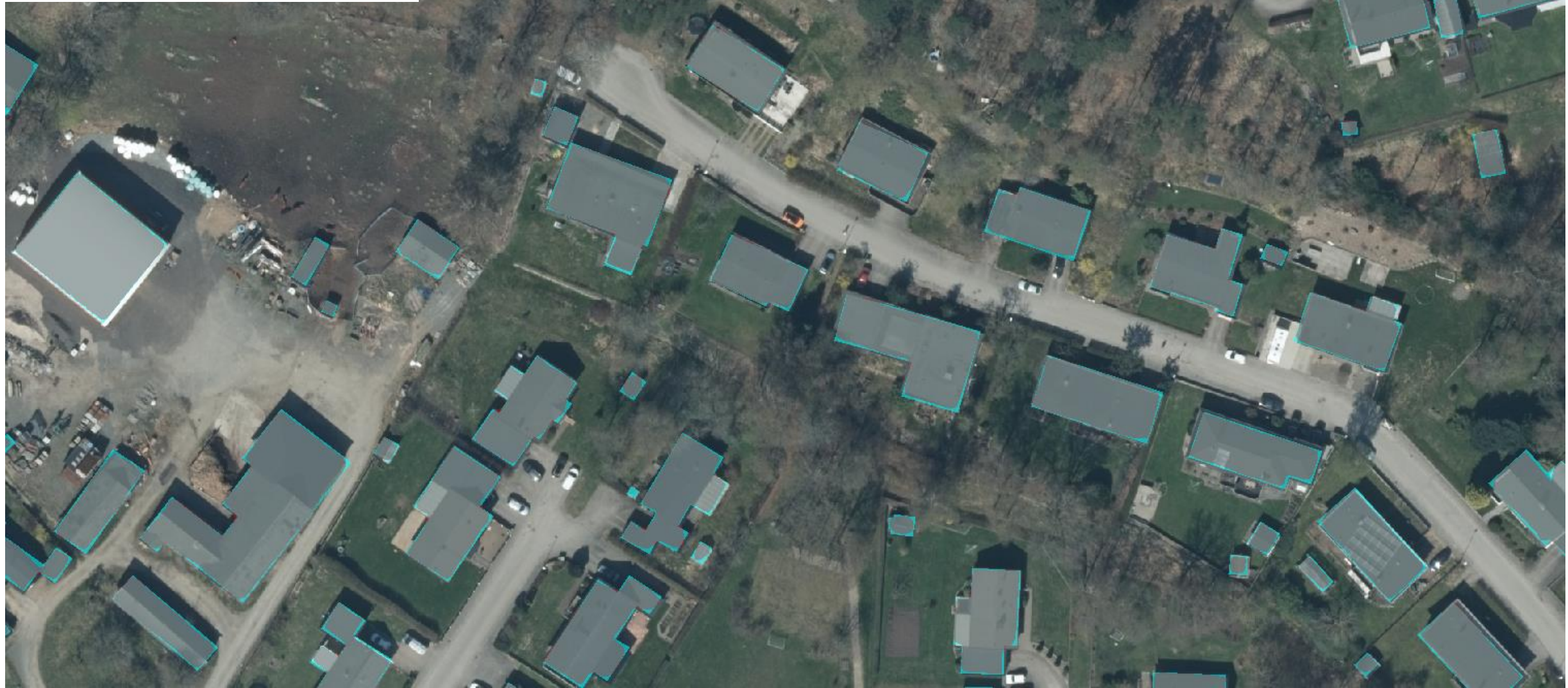
-  Object found by ML
-  Ground truth
-  Area where ML and GT match



 Object found by ML

 Ground truth

 Area where ML and GT match



# UTVÄRDERING

- Vid utvärdering av modeller använder vi 35 specifika områden som är 2,5\*2,5 km stora och kör fram byggnader över dessa. Vi fokuserar särskilt på 5 områden där vi kontrollerat att alla byggnader finns med.
- Vi beräknar statistik med hjälp av ett verktyg som heter **Compute Accuracy For Object Detection** i ArcGIS Pro
- Vi rensar bort alla detekterade byggnader mindre än 2m<sup>2</sup> innan vi beräknar statistik.
- Vi försöker också få ett visuellt intryck från områdena.
- Ca 30 kartblad (10\*10 km) har gått igenom av Flygbildstolkningen och gett återkoppling vilket lett till att vi kunnat komplettera träningsdatat.

# UTVÄRDERING

Blad	Precision	Recall	F1_Score	AP	True_Pos	False_Pos	False_Neg
T624_42_2500_Kullerod	0,8	0,78	0,79	1,02	117	30	33
T648_28_2525_Hunnebostrand	0,91	0,8	0,85	1,14	1925	182	473
T672_53_0025_Falun	0,88	0,72	0,79	1,23	2349	315	915
T684_57_0000_Ivarsmyran	0,61	0,66	0,64	0,93	35	22	18
T728_84_7500_Lovskar	0,83	0,6	0,7	1,38	464	96	311
Summa					4890	645	1750
<b>Totalt</b>	0,88	0,74	0,80				
<b>Max</b>	0,91	0,8	0,85	1,38			
<b>Min</b>	0,61	0,6	0,64	0,93			
<b>Medel</b>	0,81	0,71	0,75	1,14			
	0,12	0,08	0,08	0,18			

Precision= Andelen av de byggnader vi hittat som faktiskt är byggnader.

Recall= Andelen byggnader vi hittat av de vi borde ha hittat

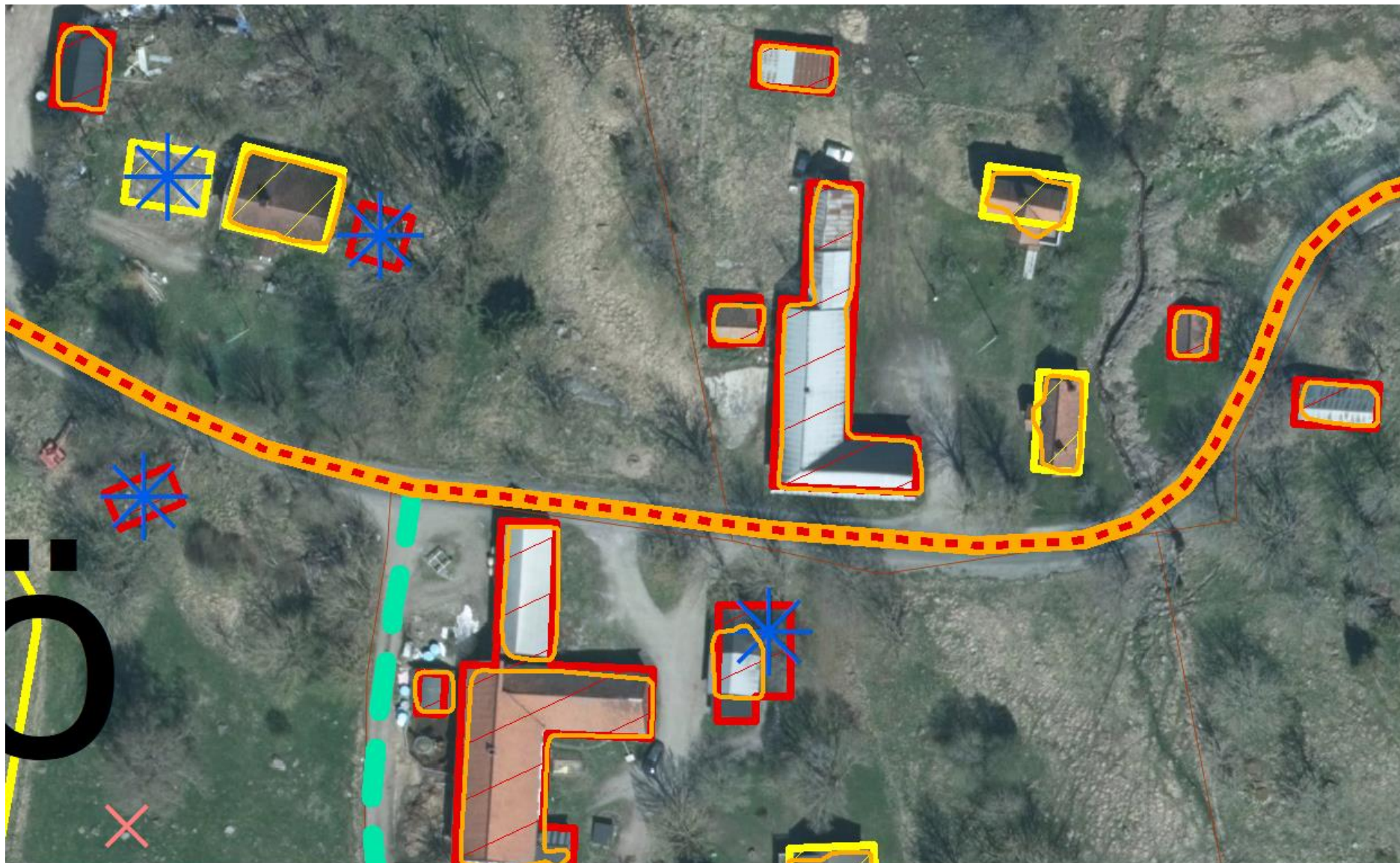
# ANVÄNDNING I FLYGBILDSTOLKNING

- Sedan 11 mars använder vi byggnader detekterade med maskininlärning som ett underlag i Lantmäteriets flygbildstolkning
- Flygbildstolkaren avgör om det är en byggnad som ska läggas in/ändras/tas bort i databasen eller inte.
- Tester indikerar att vi hittar mellan 1-2% fler byggnader på detta sätt än vid flygbildstolkning utan stöd av ML-byggnader. Dvs vi får bättre fullständighet.
- Vi tror att vi med mer träffsäkra modeller på sikt kan minska den tid vi lägger på flygbildstolkning.



# UNDERLAG ARCMAP – POLYGONER OCH DIFFERENS

- ML\_byggnad
- ML\_byggnad\_polygon
- ML\_byggnad\_punkt
- ML\_byggnad\_diff



# TEKNISK MILJÖ

- NVIDIA A10 Tensor Core GPU på Vmware-plattform
- Linux - bättre nyttjande av datorns systemresurser
- Programkod skrivs i Python med ML-ramverket PyTorch.
- FME används för databearbetningar och för att skapa träningsdata.
- Programkoden versionshanteras i Git.
- Själva produktionsmiljön ligger i en sk *container* vilket gör det enkelt att flytta den till andra datorer eftersom alla inställningar och konfigurationer man gjort följer med.

# LÄRDOMAR OCH ERFARENHETER

- Bra träningsdata är A och O – Gör stickprov för att kontrollera att det inte finns systematiska fel
- Vi tror att vi skulle ha bättre resultat om vi hade kunnat använda laserdata för att skapa ythöjdmodeller. (Samtidig laserskanning och flygfotografering ligger högt på önskelistan.)
- Tar tid att träna och utvärdera modeller.
- Tålmod – Ändra inte flera parametrar åt gången
- Kräver bra prestanda på hårdvara
- Se till att optimera hela processen så att nätverk, GPU, minne mm nyttjas optimalt

# TACK! VI FINNS PÅ...

WEBBPLATS

[www.lantmateriet.se](http://www.lantmateriet.se)

LINKEDIN

[www.linkedin.com/company/lantmateriet](http://www.linkedin.com/company/lantmateriet)

FACEBOOK

[www.facebook.com/lantmateriet](http://www.facebook.com/lantmateriet)

INSTAGRAM

[www.instagram.com/lantmateriet](http://www.instagram.com/lantmateriet)

KONTAKT

[kundcenter@lm.se](mailto:kundcenter@lm.se)

TELEFON

0771-63 63 63