

# Pythonprojekt: TIMESAT-program för bildhantering och visualisering

## Introduktion

TIMESAT är ett program för hantering av tidsserier av satellitbilder. Visualiseringsdelen av TIMESAT är idag programmerad i Matlab. För att minska beroendet av Matlab finns det en plan på att skriva om delar av TIMESAT till Python. I det här projektet börjar vi med visualiseringsdelen för enstaka satellitbilder: *TSM\_imageview* i TIMESAT (Fig. 1).

TIMESAT hemsida: <http://nateko.lu.se/timesat/>

Användarregistrering för nedladdning av TIMESAT: <https://timesat.gis.lu.se/apply/default.aspx>

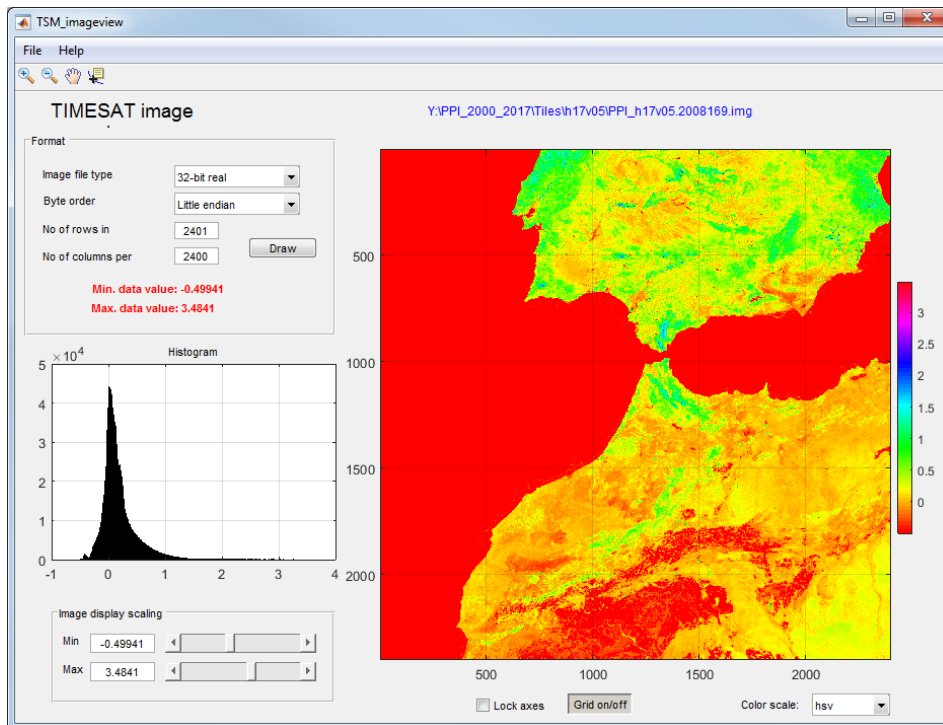


Fig 1. *TSM\_imageview*

## Uppgift

Programmet som ska skrivas ska efterlikna *TSM\_imageview* i sin funktion, men inte nödvändigtvis design. Programmet ska vara användarvänligt och innehålla följande funktioner:

- Val och inläsning av binära bildfiler; eventuellt från fillista.
- Visning av enkelbilder med olika färgtabeller.
- Zoomning och panorering i bilder.
- Visning av statistik, t.ex. min, max, histogram.

- Förändring av bildens färgskalning samt histogrammets utsträckning genom att sträcka mellan min- och max. I många scriptspråk finns denna funktion automatiskt implementerad för bilder och behöver inte programmeras. Annars använd formeln  $z\text{-bild} = (z - z_{\min}) / (z_{\max} - z_{\min})$ , där  $z\text{-bild}$  är skalat bildvärde,  $z$  är ursprungligt värde i bildfilen,  $z_{\min}$  är det minsta bildvärde som ska motsvara minimivärde i display (0 = svart), och  $z_{\max}$  är högsta bildvärde som ska motsvara maxvärde i display (t.ex. 255 = vitt, beroende på vilken skalning färgsystemet använder; kan också gå mellan 0 och 1).
- Visning av enskilt pixelvärde när man pekar på en punkt.
- Hjälptexter när man för pekaren över olika funktioner.

Det ska även finnas flexibilitet och plats för att bygga på ytterligare funktioner (t.ex. import av bilder i olika format, formatkonvertering, visualisering av tidsserier, animeringar m.m.).

## Kod

Koden ska vara väl strukturerad så att det är lätt att bygga ut och underhålla. Den ska vara väl dokumenterad på engelska och skriven på transparent sätt (gärna med explicita loopar etc. som är lätta för läsaren att förstå hellre än väldigt kompakt som är svårläst). Variabelnamn ska vara till stor del självförklarande (t.ex. "imagewindow" i stället för "xyzsdijsdfly"). I största möjliga mån ska standardfunktioner/bibliotek användas som kan implementeras både för Windows och Linux.

## Rapportering och tidsplan

Rapportering	Vecka
1. Grov designskiss: visar ungefärligt utseende, vad som ska finnas i menyerna etc.	Se sep. dok.
2. Systemskiss: Beskriver programmets struktur i form av subrutiner och interna logik.	Se sep. dok.
3. Preliminär version: Visar första implementering som visar att programmet huvudsakligen fungerar, t.ex. för visning av bilder.	Se sep. dok.
4. Slutversion: fullt dokumenterad kod med slutlig funktionalitet.	Se sep. dok.
5. Slutrapport och muntlig redovisning.	Se sep. dok.