**Imagery\_header\_transform.exe (Windows)**

**Imagery\_header\_transform.py**

(previously shift\_datum.exe)

**Release date: 2022-08-24**

**Contact**: [Joel.Haasdyk@customerservice.nsw.gov.au](mailto:Joel.Haasdyk@customerservice.nsw.gov.au)

Commissioned, but did not write this code.  
Any modification to this code will be gratefully received.

<https://customerhub.spatial.nsw.gov.au/servicedesk>

Holds this code in repository. Does not actively support this code. (at time of release)

**Status**: NOT FOR PRODUCTION  
 SUITABLE FOR TESTING ONLY

Likely to require further modification to suit specific user needs.

WARNING: Overwrite existing files UNLESS a new output directory is specified.

**Background**:

*Imagery\_header\_transform.exe* was commissioned by NSW Department of Customer Service around the year 2020 to assist with the rollout of GDA2020, and the transformation of Imagery. This utility employs [GDAL](https://gdal.org/) and [PROJ](https://proj.org/) functionality to perform this transformation by modifying the header metadata only.

Several transformation methods are possible:

* Resampling is commonly employed by software for datum transformation. However testing demonstrates that resampling is extremely inefficient, and can degrade the quality of the resulting imagery. Resampling should be avoided where possible.
* An alternative method is to update only the coordinate and datum metadata stored in the imagery header and/or auxiliary file(s).

Notes:

* Some imagery stores metadata (including in some cases redundant metadata) in software specific Persistent Auxiliary metadata (PAM) files, or ‘sidecar’ files e.g. ESRI’s .aux.xml files.   
  This utility does NOT transform PAM files. Consider instead to delete auxiliary files prior to transformation and allow other applications to recreate them where needed.
* At time of writing, to update the header information in ECW/JP2 files, an SDK licence is required. E.g. ECW JPEG2000 SDK v5 Server Read-Only End-user License available from   
  <https://www.hexagongeospatial.com/> (approx. $500 annual)
* GeoPDF are currently failing transformation tests.
* Applies Conformal plus Distortion NTv2 transformation EPSG:8447 ‘GDA94 to GDA2020 (2)’,

(or if outside those extents, then 7-Parameter Conformal EPSG:8048 ‘GDA94 to GDA2020 (1)’).

(continued next page)

**Known issues / limitations**

* Currently accepts only GDA94 and GDA2020 input (for Autodetection)
* Currently accepts only MGA projections (not GDA2020 Geographic Lat / Long).
  + Autodetect of GDA Geographic SRS fails
  + Overriding EPSG input / output to GDA Geographic fails (‘Invalid coordinates’)
* Imagery (e.g. TIFF) transforms as expected, but may or may not properly write SRS metadata.

e.g. If importing the transformed product into ArcMap, we have observed:

A TIFF transformed MGA94 -> MGA2020 may report as “Spatial Reference undefined”,   
A TIFF transformed MGA2020 -> MGA94 will be properly registered “MGA94”.

“Import product into ArcMap yields expected alignment, but

* GeoPDF currently fails to transform.
* Some imagery types (e.g. ECW, JPEG2000) require licencing to overwrite metadata info.
* Overwrites existing files UNLESS a new output directory is specified.  
  This also means that if your input file is called GDA94, your output file will retain this NAMING

**Outstanding (?) development / extensions (in addition to addressing the list of limitations).**

* Support folder and sub folder input (currently appears to recursively search input directory)
* Support relative folder input (notation *.\relative\_path\* is supposed to work, but testing fails.)
* Options for source and target srs inputs are currently supported, but not well tested.
* Apply to datasets referenced with GCPs (GeoPDFs) – currently these transform INCORRECTLY with errors at the metre-level. Characterisation of the issue is incomplete, inconclusive. Consider GeoPDF use of ‘ground control points’ (GCP)??
* Choice of [GDA94 <> GDA2020 transformation](https://www.spatial.nsw.gov.au/surveying/geodesy/transformation_methods) [7 Parameter conformal, NTv2-CPD, NTv2-CON]
* Improved selection of output CRS (currently only matches on MGA or datum)
* Support for proprietary sidecar Georeference files?

Some information is available: <https://gdal.org/drivers/raster/gtiff.html#georeferencing>

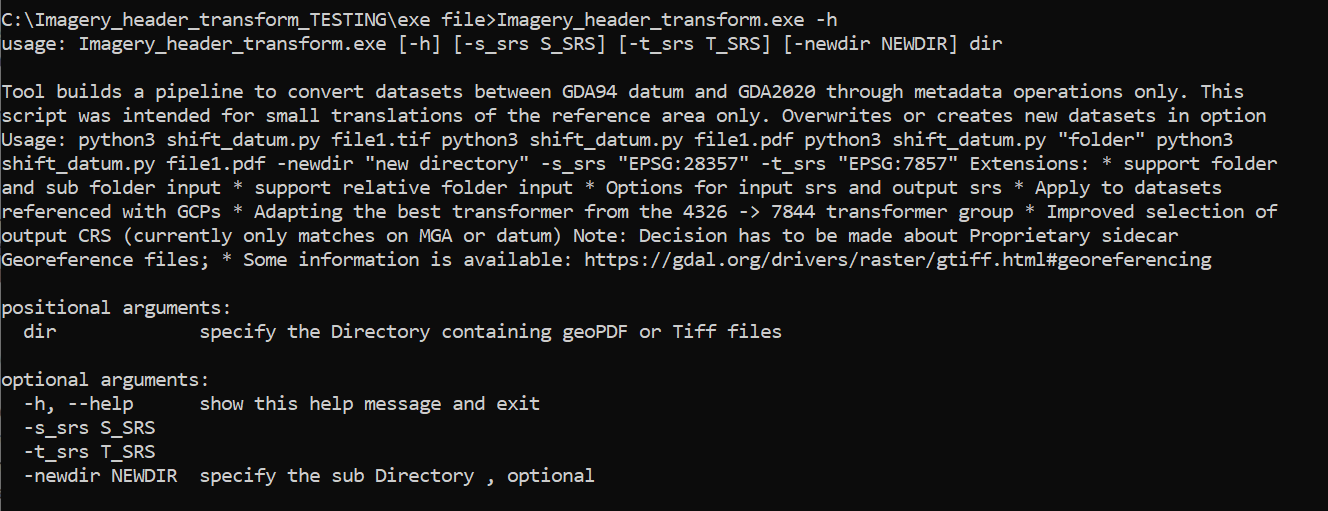
* Simplify code logic / extents checking by adding the [Custom NTv2-CPD-plus-LordHowe-Norfolk](https://portal.spatial.nsw.gov.au/portal/home/item.html?id=f39d633067ba44b6962fa99fdfcf70a0)

* Requires more rigorous requirements documentation before further development.

(continued next page)

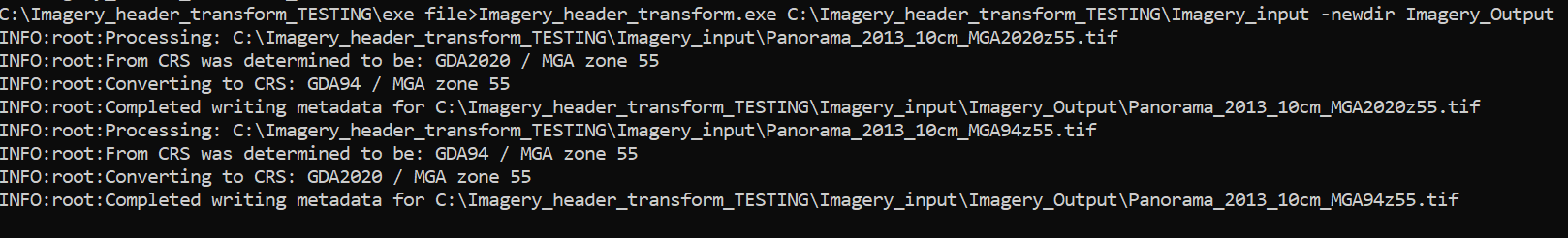
**Usage (EXE):** Note: exe is quite slow as it builds of a local python environment)

* Copy exe folder INCLUDING the ‘PROJ’ subfolder to your local drive.
* Use CMD prompt; Navigate to location of exe (or add to PATH)
* At CMD prompt provide at least the exe name, and an imagery file (or folder): e.g.
  + *Imagery\_header\_transform.exe file1.tif*
  + *Imagery\_header\_transform.exe file1.pdf*
  + *Imagery\_header\_transform.exe "folder path"*
  + *Imagery\_header\_transform.exe "folder path" -newdir "new directory" -s\_srs "EPSG:28357" -t\_srs "EPSG:7857"*
* Notes:
  + If srs are not specified, then GDA94 or GDA2020 will be autodetected and the output will be the opposite (i.e. GDA94 > GDA2020 and GDA2020 > GDA94).
  + There is NO check that SRS of the file matches SRS nominated on CMD line input.



* *s\_srs is the source Spatial Reference System*
* *t\_srs is the target Spatial Reference System*
* *newdir is an optional New Directory for output (otherwise file input is overwritten).*

This example transforms the content of a directory containing both MGA94 and MGA2020 input.



*(see next page for source code installation and usage)*

**Usage (Python source code) – runs more quickly than the EXE.**

* Assuming you have an appropriate python environment, and installed packages include: pyproj, osgeo, etc.
* Copy ‘source code’ folder INCLUDING the ‘PROJ’ subfolder to your local drive.
* Use CMD prompt; Navigate to location of *Imagery\_header\_transform*.py (or add to PATH)
* At CMD prompt, provide at least the script name, and an imagery file (or folder): e.g.
  + *Python.exe Imagery\_header\_transform.py file1.tif*
  + *Or other similar examples as given for Usage (EXE) above)*

Install notes for GDAL / PROJ  
1) Installing GDAL (to satisfy nominal need for osgeo package) (WINDOWS)

* + <https://opensourceoptions.com/blog/how-to-install-gdal-for-python-with-pip-on-windows>
    - (Download the correct .whl file as indicated...
    - e.g. GDAL-3.4.3-cp39-cp39-win\_amd64.whl for Python 3.9 and so on)

2) If (1) above results in proj.DB errors (Version mismatch):

* + Note: Original code committed at 20220825 is hardcoded to look for a local proj.DB
  + (Probably because this code was created with the intention of packaging into EXE??)
  + It would probably be best to re-code that (or investigate whether a local proj.db is necessary at all), but for a quick install as-is, copy the contents of the proj directory (i.e. the proj.db etc) which came with the GDAL install above.

e.g. C:\Users\%USER%\AppData\Local\Programs\Python\Python39\Lib\site-packages\osgeo\data\proj

into the proj folder in this Project.

3) Run the code with the parameters indicated above.