



# Notes to accompany the South Launceston Landslide Map Series

Author: C. Mazengarb  
Date: 27/05/2025  
Email: [info@mrt.tas.gov.au](mailto:info@mrt.tas.gov.au)  
Website: [www.mrt.tas.gov.au](http://www.mrt.tas.gov.au)

REPORT No.: TR41



Geological Survey  
Technical Report 41





# **Geological Survey Technical Report 41:**

## Notes to accompany the South Launceston Landslide Map Series

by  
C. Mazengarb

Cover: Partially healed shallow earth slides in colluvium derived from weakly cemented Paleogene-Neogene Launceston Group alluvial sediments, between Evandale and White Hills (photo credit: N.J. Roberts).

While every care has been taken in the preparation of this report, no warranty is given as to the correctness of the information and no liability is accepted for any statement or opinion or for any error or omission. No reader should act or fail to act on the basis of any material contained herein. Readers should consult professional advisers. As a result the Crown in Right of the State of Tasmania and its employees, contractors and agents expressly disclaim all and any liability (including all liability from or attributable to any negligent or wrongful act or omission) to any persons whatsoever in respect of anything done or omitted to be done by any such person in reliance whether in whole or in part upon any of the material in this report. Crown Copyright reserved.



# Tasmanian Government

---

Mineral Resources Tasmania

[www.mrt.tas.gov.au](http://www.mrt.tas.gov.au)

Department of State Growth

---

## Notes to accompany the South Launceston Landslide Map Series

by C. Mazengarb

*Geological Survey Branch - Mineral Resources Tasmania*

---

### CONTENTS

1.0 PURPOSE .....	4
2.0 METHOD .....	4
3.0 AVAILABILITY .....	4
4.0 CAVEATS FOR USE.....	4
5.0 MAP 1. LANDSLIDE INVENTORY .....	4
5.1 Landslide Classification .....	5
5.2 Damage .....	6
6.0 MAP 2. GEOMORPHOLOGY.....	6
6.1 Geomorphological Setting .....	7
7.0 MAP 3. SIMPLIFIED GEOLOGY .....	7
8.0 MAP 4. SLIDE SUSCEPTIBILITY.....	8
9.0 CONCLUSION .....	8
10.0 ACKNOWLEDGEMENTS .....	8
11.0 REFERENCES .....	9

## 1.0 PURPOSE

This document accompanies a set of thematic maps addressing regional landslide susceptibility and zoning for an area between Evandale and as far north as Launceston CBD. The data contained in this set provides the information from which planning zones can be derived. The technical information contained is primarily aimed at the geotechnical community, who may undertake site investigations using the data as reference material. On previous maps in this series, text was placed on the maps themselves but is here consolidated into a single document to simplify map production, to more fully document the methodology and to provide the modelling code.

## 2.0 METHOD

Landslide zoning in the Tamar Valley was first undertaken by the then Department of Mines in the 1970s (Stevenson, 1978) driven by alarming damage to neighbourhoods as the area was progressively developed. While the quality of this mapping was generally very good for its time, new methods, information sources and technology have been employed to make a substantial revision. In 2003 a new methodology was developed for this and other areas of Tasmania as described by Mazengarb and Stevenson (2010). A further modification has been made since the earliest maps of this series were released (post 2007), to conform as much as possible to the publication of the Australian Geomechanics Society guidelines for landslide zoning (AGS, 2007a,b). In addition to the reasons given above, the northern extent of this study area revises the overlapping part of the 2005 Launceston map series to take advantage of subsequent LiDAR datasets that significantly improve our recognition of surface morphology and provide a more accurate base for computer modelling.

## 3.0 AVAILABILITY

Information in this report is in the public domain and anyone is free to use it provided they read and understand the purpose and limitations and provide appropriate acknowledgment of the original data source.

The information is provided in various formats:

- PDF map images obtained from the MRT website
- In a web browser accessing the MRT and LIST web map viewers
- In GIS software accessing downloaded raster and vector files or as web-services.

Full details on how to access this information is available from the MRT website ([www.mrt.tas.gov.au](http://www.mrt.tas.gov.au)) or by contacting MRT directly.

## 4.0 CAVEATS FOR USE

The following caveats apply to the maps:

- The hazards identified are based on imperfect knowledge of ground conditions and models to represent our current understanding of the landslide process. As this knowledge improves, our perception of the hazard, and the depiction on the map, may also change.
- These maps can be used as a guide (or flag) to the need for specific assessment in potential hazard areas.
- Planning decisions should not be made solely on the basis of the zones delineated on the map.
- Scale limitations of the data should be considered at all times, as exceeding this limit could lead to inaccurate decisions about the hazard.
- Site specific assessment of landslide hazard and risk should be undertaken by suitably qualified and experienced practitioners in the fields of engineering geology and geotechnical engineering.
- Practitioners undertaking site specific assessments should read the map text and associated documents to obtain a thorough understanding of the methodology and limitations of the maps.
- Areas where no susceptibility or hazard is shown can still have issues with slope instability.
- Anthropogenic influence on slopes cannot be predicted and the occurrence of slope instability resulting from the influence of human actions is specifically excluded from these maps.
- The identification and performance of cut and filled slopes have not been specifically considered in map production, and their scale is such that they often cannot be resolved on the maps. The presence of such slopes should always be considered in site specific assessments.

## 5.0 MAP 1. LANDSLIDE INVENTORY

Landslide data shown on this - and associated maps - is derived from a statewide landslide database administered by Mineral Resources Tasmania (MRT), and made available for public consumption. Additional information may exist for many of the features shown, and this can be obtained by contacting MRT directly. Data stored within the landslide database is sourced from mostly internal and some external sources. Unfortunately, most geotechnical reports that are required for residential and subdivision development applications (including 'soil tests' for foundation and waste-water purposes) are not generally available to MRT.

As further information is added to the landslide database, the map view described here will become out-of-date. As such, this map supersedes the overlapping part of the previous map by Latinovic and Latinovic (2005). Revisions are typically based on improved base information but ultimately rely on professional judgements by the author(s). It is important to state that this latest iteration has not only modified the shape and position of some previously mapped features, but others have been removed as the evidence for their existence has been re-evaluated. It is likely that there are many unrecognised landslides or human-modified forms in the landscape that may be revealed after these maps are published. It is therefore very important that the live landslide database is always consulted when using these maps.

This landslide inventory is regarded as a fundamental underpinning layer for the development of susceptibility and hazard maps. As part of the compilation process, all data shown on the map has been checked by MRT geologists to ensure that it meets the necessary minimum standards for landslide description. Landslide data that have not passed MRT quality standards are not shown on the map but are kept in the database as a closed file. All reasonable effort has been made to ensure the landslides are accurately located and attributed. It should be appreciated that most landslide boundaries are approximate only, and largely based on the interpretation of aerial photos or airborne laser scanning (LiDAR) with limited field mapping. The mapping has utilised aerial photography dating back to the 1940s, enabling more accurate mapping of features within areas that have since been modified by human activity. Many small landslides may not have been recognised in this mapping exercise, although those that are in the MRT database, that are too small to depict at this map scale, are shown as point features.

Depictions of individual landslides and internal landslide features, on the Inventory Map (Map 1) and the Geomorphology Map (Map 2), are representations of the morphological features that currently exist in the landscape - rather than an estimation of the original, uneroded morphology. As landslide features are eroded over time their shape will change. For example, with ongoing erosion, landslide headscarps will tend to become broader and slopes will reduce; while landslide toes may become flatter and less apparent or may be trimmed by more active erosion (by the ocean or a stream) and so become steeper. The outlines on the Inventory Map (red on the Geomorphology Map) show the total affected area of each landslide, or landslide zone. The internal features within these affected areas are depicted on the Geomorphology Map, i.e. headscarp (or main scarp), and features within the displaced mass - landslide bench, internal minor scarps, erosional scarp at toe - and linear depressions.

Linear depressions are the probable representation of the former head/heel of back-tilted blocks, transverse cracks, or crown cracks above the headscarp, and are often defined in the landscape by internal drainage. These features have been depicted on the Geomorphology Map where scale allows, however the MRT landslide database contains all internal features where they have been mapped.

Some landslides are disguised by their formation in a position that has exposed their toes to significant erosion at some point (e.g. by past higher sea levels) and often have some of their displaced mass removed. In other situations, it was found that some of the features previously mapped as landslides were in fact differential erosion features, and many of the 'possible' landslides on these maps are those where it was not certain that differential erosion could be ruled out. It should be noted that hummocky ground caused by shallow soil creep has not been included in this mapping programme.

### 5.1 Landslide Classification

A landslide is defined as a downslope movement of a mass of rock, debris or earth. This broad definition includes a variety of failure modes and is not only limited to slide-type failures. However, ground subsidence and collapse are excluded. The material involved may be either 'rock' (a hard or firm mass that was intact and in its natural place before initiation of movement), or engineering 'soil' (an aggregate of solid particles either sediment/transported or formed by weathering of bedrock). 'Soil' is further divided into 'debris' (more than 20% of material coarser than 2 mm) and 'earth' (more than 80% of material finer than 2 mm).

There are five kinematically distinct types of landslide movement recognised in Tasmania: fall, topple, slide, flow and spread (Cruden and Varnes, 1996).

1. Falls and Topples - Falls refer to detachment and very rapid movement (falling, bouncing and rolling) of material from a steep slope. Toppling failures are distinguished by forward rotation about a point below the centre of gravity of the displaced mass. Landslides of these types occur on steep slopes or cliffs. Both falls and topples occur in restricted areas within the mapped area, particularly gorges and quarries associated with Jurassic dolerite and Cenozoic basalt.
2. Slides - (in a more restrictive sense of the term 'landslides') are movements of material along recognisable shear surfaces or zones. The shear surface may be curved and concave (rotational slides) or roughly planar (translational slides). Within the study area, both are common as well as transitional types, including slides that develop into flows. These are commonly developed in soft or weathered Paleogene sediments (Launceston Group) and basalts.

3. Flows - Flows refer to a spatially continuous movement of material where inter-granular movement predominates over shear surface movement. Within the study area both debris flows and earth flows occur. Flows can develop as secondary movements in the toe area of slide-type movement within Paleogene basalts and sediments.
4. Block Slides - These are a special case of transitional slides that are likely to include other styles of movement and have a complex history. In several places in the Tamar Valley, outside of this map sheet, large blocks of Paleogene basalt have separated from the main mass and moved downslope by sliding on the underlying soft Paleogene sediment (e.g. Craighburn Rocks near Hillwood). The mechanism of movement is probably partly block spread, as defined by Cruden and Varnes (1996) in the early stages, and transitions into block slide, and block topple, as it moves downslope. It is possible that some outliers of basalt could be displaced blocks of this type, but this is often very difficult to prove. Block spreads are typically extremely slow-moving features
5. Complex Landslide Zones - Many of the larger landslide features shown on this Landslide Inventory map are in fact complex Landslide Zones that involve a number of separate landslide movements (refer to the Geomorphology Map) and have been classified on their dominant landslide movement type.

Landslides shown on this map are also classified based on their level of activity:

- Recent or Active - Landslides that are currently moving or have moved in recent times (i.e. since European settlement). Landslide features (head scarp, flanks, toe and related cracks) are commonly fresh and easily recognisable. Damage to infrastructure and property is usually visible.
- Activity Unknown - This category refers to landslides that have no evidence of recent (European era) movement and includes those that have been significantly modified through erosional processes. In accordance with Cruden and Varnes (1996) this category will include a range of inactive features, such as dormant, abandoned, stabilised and relict. This also includes landslides that were previously referred to as ancient, or fossil, by MRT geologists but this classification is now regarded as obsolete and perhaps incorrect. Importantly, it implied to end-users that these are stable features which, in some instances, has been proved otherwise. Therefore, the potential for reactivation should be assessed on a case-by-case basis.

- Possible - Mapped landscape features that have several of the characteristics of a landslide but due to significant weathering, or modification by urban development, it is difficult to be confident that they are indeed landslides. For this reason the activity of these features is unknown.

## 5.2 Damage

Previously documented or observed damage from landslide movement to buildings, roads, railways and other urban infrastructure is routinely stored in the landslide database. Such information is presented on the inventory map. However, there are likely to be many more instances that have not been reported to MRT.

## 6.0 MAP 2. GEOMORPHOLOGY

This geomorphic map completes the remapping of the main landslide susceptible areas of the Tamar Valley, in addition to those previously published in 2005 and 2013. The mapped features represent both individual geomorphic units and structural aspects of the landscape that assist in understanding its evolution and in particular land instability.

A digital terrain model (DTM) has been constructed by MRT that forms the base layer for the map. This layer combines LiDAR datasets and, where gaps exist, lower resolution data to create a complete coverage of the area. The DTM is a 10 m resolution slope raster symbolised using the approach of Demek (1972), but customised with stability thresholds determined by reference to the local underlying geology.

Manually mapped units are depicted as lines, polygons and points to represent a range of natural and artificial features whose relationships are included in the associated diagrams and legend on the map. The process of mapping made substantial use of detailed airborne laser scanning (LiDAR) surveys assisted by aerial and satellite photography with digitising at scales at much higher resolution than the scale of this map. Mapped morphological features are generally considered to be accurate to 5 m or better.

Older aerial photo surveys archived by Land Tasmania (e.g. 1946, 1952 and 1975 epochs) were also utilised in the mapping, including the creation of orthophotos. Older photos have the advantage of allowing the identification of features subsequently obscured by urban development and roads or land degradation in intensively farmed areas. Limited field checking accompanied the mapping, and the accuracy of the units recognised is reflected in the classification contained in the legend.

Natural features mapped include landslides and their internal features where discernible, as a special category of importance, in addition to fluvial and coastal landforms. Emphasis has been placed on mapping various surfaces, such as fluvial terraces, that have relative age significance. This information is used to constrain the age of the landscape, and in particular the landslides. Such information in turn can allow for an estimation of likelihood of recurrence.

Artificial features mapped include quarries, major road cuts, areas of landfill and reclamation.

Data models for these various layers are contained in Appendix 1.

### 6.1 Geomorphological Setting

The mapping area is situated at the southern end of the Tamar Valley, including the southernmost part of the Tamar Estuary and the confluence of the South and North Esk rivers. The estuary is a drowned river valley (of the 'ria' type) that most recently formed because of sea level rise following the last glacial period. During the many glacial periods in the Quaternary, when sea-levels were substantially lower, the river flowed many tens of kilometres beyond the current river mouth, cutting a deep channel (as imaged in bathymetry between Low Head and Beauty Point) along its course.

There are a range of geomorphological features expressed in the broader Tamar area, many of which reflect the underlying geological structure. The Tamar Valley is situated within the central valley of a NNW-SSE oriented asymmetric graben system, active in the Late Cretaceous to Paleogene. Gently SW-tilted fault blocks, with some western blocks tilted to NE, have given rise to a sub-parallel system of ridges and valleys formed in mainly Jurassic dolerite, with Paleogene Launceston Group and Quaternary fluvial sediment infill along with associated Paleogene basalt flows.

In late Cenozoic times, regional uplift and inversion of the landmass commenced triggering significant rejuvenation landscape processes. This uplift caused streams to lower their base levels through incision, with down-cutting occurring progressively in a headward fashion. The junction between youthful, incised stream segments, and mature, broader waterways is commonly expressed as knick points. The speed of headward migration of the knick point is strongly controlled by the resistance of geological materials to erosion. In the South Esk River (at the head of the Cataract Gorge) the nick point has only migrated 9 km because it is cutting through the doleritic Blackstone Hills. Above this knick point there is a perched landscape (above 140 m ASL) containing an extensive regional peneplain (the Woodstock Surface of Nicolls, 1960) and associated river terraces that are cut

slightly within it. In contrast, the North Esk and Rose Rivulet have knick points that have migrated much further (~24 km for the Rose Rivulet) with a long history of down-cutting punctuated by episodic terrace formation.

Within Rose Rivulet there is a series of benched landscapes below the Woodstock Surface. These benches are typically formed on sub-horizontal, harder conglomerate beds occurring within the claystone-dominated Launceston Group.

A further consequence of the headward incision of the Rose Rivulet is that it is actively pirating the South Esk Catchment, including an impressive example at the head of Boyes Creek, 5 km east of Evandale. Most importantly, with regard to slope-stability, where 'rapid' incision of streams cutting into weak Launceston Group material is occurring, the valley walls have steepened to a point where they are no longer stable. Over time these slopes slacken to a particular threshold angle depending on the materials involved. However, the presence of stronger, more erosion-resistant capping units on valley walls, such as basalt flows and intraformational conglomerates (within the Launceston Group) locally prevents the slopes slackening to a stable value, further adding to slope instability. Additionally, other factors, such as local hydrogeological conditions, the presence of springs, along with various human factors (e.g. inappropriate waste-water operation and surface drainage, vegetation removal, cuts and fills) all contribute to the potential for instability of parts of the landscape.

### 7.0 MAP 3. SIMPLIFIED GEOLOGY

A simplification and adjustment of the existing 1:25 000 scale geology map of Launceston (Forsyth and Calver, 2005), in combination with additional mapping by the author to the south, is presented. This information supports our understanding of the surface, and near-surface geology and the material properties involved. The layers created in this process are datasets used in the associated landslide modelling.

The layers created consist of:

- Bedrock geology (Permian to Cenozoic) somewhat amalgamated into major units and where reasonably confident, extended beneath surficial units
- Surficial geology (late Cenozoic), also shown on the geomorphology map
- Alteration units (duricrusts and weathered zones) derived without modification from previous maps
- Artificial deposits such as significant road embankments, landfill and reclaimed areas, also shown on the geomorphology maps
- Landslides, also shown on the inventory, geomorphology maps and relevant susceptibility maps

Adjustments to existing mapping, and the extension to the south, made substantial use of detailed airborne laser scanning (LiDAR) surveys, recent and historic aerial and satellite photography, with digitising at scales at much higher resolution than the published scale of this map. Mapped features with a morphological expression are considered to be accurate to 5 m or better in most cases. In other areas the mapping accuracy may be less accurate.

A considerable effort was made to compile drillhole and geotechnical data into MRT databases. This information, along with sample and outcrop locations, provide the user with known properties and mapping constraints. Much of this compiled information is derived from MRT sources and its predecessor, the Department of Mines. Some of the MRT data presented may not be readily available from this organisations' website, or as web services, in which case the user can request it directly from MRT.

Other adjustments were based on recently acquired geophysical information, in conjunction with subsurface data and surface outcrops, to substantially modify the fault pattern from previous mapping. This interpretation is supported by showing the reduced level elevations of the base of the Launceston Group (mostly resting on Jurassic dolerite) encountered in a small number of drillholes, and by depicting the orientation of dipping surfaces on areas of dolerite landscape that are all linked to the development of the Tamar Graben.

A limited field checking exercise accompanied the mapping, with outcrop observations depicted on the map and loaded into MRT databases.

## **8.0 MAP 4. SLIDE SUSCEPTIBILITY**

This map provides a predictor of where in the landscape landslides and earthflows (excluding rockfall) could occur. Susceptibility classes have been defined in a simplistic manner by identifying source areas above nominated slope thresholds for the various geological units mapped. These threshold values ultimately rely on professional judgement based on field observations (our own and from previous reports), desktop analysis of landscape models and consideration of laboratory measurements of material properties. Details of the modelling methodology presented here is slightly modified from Mazengarb and Stevenson (2010) and attempts to satisfy the AGS (2007a,b) guidelines as much as reasonably possible. The confidence in the susceptibility modelling is highest for the Launceston Group sediments because of the richness of the available data. For the other units, where there is much less data, the values chosen are largely based on consistency with other mapping areas as indicated.

Of particular note, landscapes underlain by Launceston Group sediments contain about 80% of the landslides in

the study area. Of the remaining geological units, landslides are much less common, in part because they reside in more mature landscapes (and lower slopes) while others, such as the Tasmanian (Jurassic) Dolerite Formation, are clearly more resistant to mass-wasting.

Several landslides have been discovered within the colluvial deposits that occur on the flanks of Jurassic dolerite and Cenozoic basalt-capped hills, but these have not been consistently mapped over the study area. While they have hummocky topography suggestive of large failures, some of these are probable periglacial features. The stability of these is significantly influenced by local groundwater conditions and the physical properties of the underlying geological materials.

Setback (regression) and runout values are assigned to allow the modelling method to identify the uphill and downhill extent respectively to which a potential failure could occur. The inputs to the modelling are the landslide mapping contained in landslide inventory, digital terrain models (DTMs), and simplified geological maps. In addition, a limiter in the code prevents unreasonably long runout zones being created. The final output incorporates various smoothing methods to ensure that a clean cartographic output has been produced to assist translation of these susceptibility features into planning tools.

This map supersedes earlier mapping by MRT, including the pre-2001 Tamar Valley - Advisory Landslide Zoning and the Launceston map by Mazengarb (2005) where it overlaps.

## **9.0 CONCLUSION**

The susceptibility modelling was undertaken in Python, using the code shown in Appendix 2. A series of thematic maps have been produced for the Evandale and Launceston (part) areas covering portions of the Launceston and Northern Midlands local government areas. This information completes a remapping of the original Tamar Valley advisory zoning of the 1970s, and a minor revision in the 1990s, when the information was digitised into GIS format. The new maps also supersede 2005 generation mapping where it overlaps.

This document, and other information on this map series or Tasmanian landslides in general, is freely available from the MRT web site or by contacting the agency directly.

## **10.0 ACKNOWLEDGEMENTS**

The author wishes to thank past and present colleagues at MRT who have assisted this compilation in various ways. Claire Kain and Nick Roberts assisted with field work and along with Andrew McNeill have reviewed the manuscript. Jo-Anne Bowerman and Genevieve Simard assisted with cartographic presentation.

## 11.0 REFERENCES

- AGS. 2007a. Guideline for Landslide Susceptibility, Hazard and Risk Zoning for Land Use Planning, *Australian Geomechanics*, 42:13- 36.
- AGS. 2007b. Commentary on Guideline for Landslide Susceptibility, Hazard and Risk Zoning for Land Use Planning. *Australian Geomechanics*, 42: 37- 58
- Demek, J. 1972. *Manual of Detailed Geomorphological Mapping*. International Geographical Union Commission on Geomorphological Survey and Mapping.
- Cruden, D. M. and Varnes, D. J. 1996. Landslides Types and Processes, In: A. K. Turner and R. L. Schuster (eds.), *Landslides: Investigation and Mitigation*. National Research Council, Transportation Research Board, *Special Report*, 247, p36 - 75. National Academy Press, Washington, D.C.
- Latinovic, N. and Latinovic, M. 2005. Map 1, Launceston – Landslide Inventory. *Tasmanian Landslide Hazard Series (1:25 000)*. Mineral Resources Tasmania, Dept. of Infrastructure Energy and Resources, Hobart.
- Mazengarb, C. 2005. Launceston, map 4 - Potential rock-fall hazard. *Tasmanian Landslide Hazard Series (1:25 000)*. Mineral Resources Tasmania, Dept. of Infrastructure Energy and Resources: Hobart.
- Mazengarb, C. 2005. Launceston, map 5 - Potential landslide hazard. *Tasmanian Landslide Hazard Series (1:25 000)*. Mineral Resources Tasmania, Dept. of Infrastructure Energy and Resources: Hobart.
- Mazengarb, C. and Stevenson, M. D. 2010. Tasmanian Landslide Map Series: user guide and technical methodology. *Tasmanian Geological Survey Record*, 2010/01. Mineral Resources Tasmania. 44p.
- Mazengarb, C. and Stevenson, M. D. 2011. Evaluation of the Tasmanian North-west Coast landslide map series against the AGS 2007 landslide zoning guideline. *Australian Geomechanics*, 46(2): 219-233.
- Nicolls, K. D. 1960. Erosion Surfaces, River Terraces, and River Capture in the Launceston Tertiary Basin. *Papers and Proceedings of the Royal Society of Tasmania*, 94:1- 12.
- Sutherland, F. L., Graham, I. T., Forsyth, S. M., Zwingmann, H. and Everard, J. L. 2006. The Tamar Trough Revisited: Correlations between Sedimentary Beds, Basalts, their Ages and Valley Evolution, North Tasmania. *Papers and Proceedings of the Royal Society of Tasmania*, 140:49- 72
- Selkirk-Bell, J. M. and Mazengarb, C. 2005. Launceston, Map 2 - Geomorphology. *Tasmanian Landslide Hazard Series (1:25 000)*. Mineral Resources Tasmania, Dept. of Infrastructure Energy and Resources: Hobart.

# APPENDIX 1

## GIS DATA MODELS

## GIS data models

It was necessary to create a series of GIS layers for the Tasmanian Landslide Series, each with an original data model. For all but one layer, the data was not part of an existing entity at MRT and was created from scratch. However, the exception was the 1:25k geology layer that required modification to be fit for purpose for this project. This appendix provides a description of these layers and their purpose. The data model was largely developed in time for the NW area mapping in 2010 with subsequent minor modifications in the Tamar Valley including Evandale. Earlier mapping in Launceston, Hobart and Glenorchy differs and will be eventually reclassified to match. The intention at the time of writing is that these layers will be compiled into a seamless state-wide coverage that is made available via a web service allowing the Geohazards staff to incorporate other areas in a timely manner independent of formal cartographic map production.

There are 5 general themes containing several layers, some of which are shared between each theme. Each of these themes are broadly described in the main text of the document:

- Landslide inventory
- Geomorphology
- Geology
- Landslide (slide) susceptibility

### Landslide Inventory

Purpose: A theme to show the location and knowledge of landslide features at the time of capture.

Components: The landslide inventory contains four spatial layers derived from the publicly available MRT landslide database as web services and download.

- Landslide points symbolized to show age and movement style concepts.
- Landslide polygons that have been simplified (see below) to indicate knowledge of the age of activity (either Recent viz post-European settlement, or activity unknown)
- Landslide damage point locations where knowledge is in the public domain.
- Landslide damage polygons for features that are large enough to show on the map, particularly buildings and roads. Note that these polygons deliberately overlie the corresponding point features to simplify the map.

Discussion: The map provided is a static image and with time will become outdated as more landslide records are added. While the information is in the public domain, the map is currently not able to be automatically updated as manual geoprocessing of the landslide polygon layer is required. However, it may be possible to generate these features as a materialised view, live from the Oracle database, if consistency of data entry is maintained.

### Geomorphology

Purpose: A theme to provide an understanding of landscape components and processes including mass movement.

Components: There are 6 spatial layers that comprise the mapped geomorphology.

1. Quarries and Mines: These are point features derived from the MRT Deposits database where a quarry or mine has been registered. Unfortunately it is not complete and some of the locations and attributes need amending.
2. Artificial polygon features: These contain anthropogenic features such as road cuts, embankments and quarries. The layer is placed above natural geomorphic features in the GIS view recognizing that they are usually younger.
3. Artificial line features: These contain anthropogenic linear features, namely cuts and levees. Symbology is controlled by the feature type and direction of line digitization in the case of cuts
4. Geomorphic point features: These are point features to capture springs, knickpoints and structural surface orientations. The symbolization is controlled by the feature type and orientation and labels by attribute fields
5. Geomorphic line features: These are line features representing a variety of natural features including breaks in slopes, lineations, etc. In some instances these lines coincide with the boundaries of polygons such as terraces, whereas others are independent features. Some of these features are also shown in the geology layer but symbolized differently.
6. Geomorphic polygon features: These are polygon features representing a variety of natural surficial features, both surfaces and deposits. Most of these features have a direct match to those in the surficial geology layer but symbolised differently.

The layers listed above have some common attribute field concepts:

**Feature type:** the fundamental classification that mostly controls the symbology. The field does not contain a concept of attribute confidence unlike the MRT 25k geology layer as a separate qualifier field is included to capture this concept.

**Qualifier:** A measure of the confidence of the feature (feature type) recognition.

Code	Description	Label suffix for polygon features
Definite	Definite as observed in field or photos	
possible	Possible based on circumstantial evidence	?
definite but conceal	Definite but concealed	
possible and conceal	Possible and concealed	?

**Label:** The labels used on map that are constructed by combining the feature-type code with the qualifier suffix.

**Description:** A standard description of the Feature type field

The following concepts as defined by Fairbridge are applied wherever possible to the geomorphic features mapped.

### Forms

Code	Description
marine	Marine form seawater agent
fluvial	Fluvial form stream water agent
aeolian	Aeolian form wind agent
denudative	Denudative form gravity and water agent
anthropogenic	Anthropogenic form human agent
glacial	Glacial form glacier agent
nival	Nival form snow agent
cryogenetic	Cryogenetic form frost agent
limnic	Limnic form lakewater agent
organogenetic	Organogenetic forms plants and animals
karstic	Karstic form surface and underground water agents
tectonic	Tectonic form endogenic force (agent)

### Processes

Code	Description
constructive	constructive
destructive	destructive
destructive-solution	destructive by solution
const destruct	constructive and destructive

### Agents

Code	Description
Seawater	Seawater
stream water	Stream water
wind	Wind
man	Man
gravity and water	Gravity and water
glacier	Glaciers
lakewater	Lake water
surface underground water	Surface and underground water (karstic)
frost	Frost
snow	Snow
plants animals	Plants and animals
endogenic	Tectonic processes (forms)

## Geomorphic polygons

The following units with their descriptions are recognised:

Code	Description
alluv terrace 10	Alluvial terrace approximate 10m elevation
alluv terrace 15	Alluvial terrace approximate 15m elevation
alluv terrace 20	Alluvial terrace approximate 20m elevation
mar terrace 5	Marine terrace approximate 5m elevation
mar terrace 10	Marine terrace approximate 10m elevation
mar terrace 15	Marine terrace approximate 15m elevation
mar terrace 20	Marine terrace approximate 20m elevation
backshore	Backshore
coastal plain	Undifferentiated coastal plain
Holo coast deposits	Undifferentiated Holocene beach deposits
Pleist beach deposits	Undifferentiated Pleistocene Beach Ridges and Dunes
mar terrace	Undifferentiated marine terrace
backswamp	Backswamp
sea stack	Sea stack
aband sea stack	Abandoned sea stack
aband coastal scarp	Abandoned coastal scarp
alluvial terrace 5	Alluvial terrace approximate 5m elevation
coast cliff	Modern coastal cliff
salt marsh	Salt marsh
flood plain	Modern floodplain
river	River
alluv terrace	Undifferentiated alluvial terrace
vall floor	Valley floor gently sloping sheet wash and alluvium
perch vall floor	Perched Valley Floor
aband perch valley	Abandoned Perched Valley
reg surface	Regional Landscape Surface
elev surface	Elevated Surface, origin uncertain
struct surface	Structurally controlled surface
hillslope	Undifferentiated hillslope
bench	Bench, midslope
coll footslope	Colluvial footslope
alluv fan	Alluvial fan
unk	Unclassified
landslide	Landslide affected area
headscarp	Landslide headscarp
disp mass	Landslide displaced mass (excluding flows)
minor scarp	Landslide minor scarp
landslide trim	Trimmed (eroded) part of landslide
landslide bench	Landslide bench
flow	Earth or debris flow displaced mass

## Regression code (Python 3.7)

```
## Author: Colin Mazengarb
## Purpose of Script: This script calculates a setback area above a slope that potentially could be affected if a landslide
was to form.
## This version works from the python console of QGIS (3.16 or higher) as all of the required modules should be
installed with QGIS.
# All input raster file formats are tifs. They must be of identical extent, same cell size and fully aligned!
## Input and output locations are specified in OutputWorkspace and WorkSpace variables
## "lsrc" = the name of the source grid. Shall consist of source areas = 1, non source areas = 0 or null. This requires
a preceding operation whereby a slope threshold value is assigned to a geology layer (each geological unit can have a
unique threshold). This value is compared to a slope DTM to create the true or false values.
## "dem" = the name of the digital elevation model used in the model; any floating point values or null.
## "aspect" = the name of the aspect grid used in the model; positive floating-point numbers ranging from 0 to <360,
or -1 (horizontal).
## "beta" = the name of the Beta angle grid: integers,  $0 < \text{Beta} < 90$ . Beta is the angle of regression from a source raster
linked to the geology layer
## "regress" = the name of the output regression grid; shall consist of positive integers representing distance (multiples
of the cell size) from the source cell. Where a regression cell is identified in more than one regression path, the lower
value will take precedence. A limit on the amount of regression can be performed as a subsequent query if needed
## DEFINITION OF (Most) TEMPORARY VARIABLES USED IN THE ARRAY LOOPING PART
## X= coordinates of analysed cell
## dX= increment of length for each analysis
## Y= coordinates of analysed cell
## dY = increment of length for each analysis
## elevSource = elevation of source cell (I,J)
## elevCell = elevation of cell (X,Y)
## elevBetaLine = elevation of Beta (regression) line
## elevDiffBeta = change of elevation for each regression line progression for a given regression (Beta) angle
## AspectUphill = opposite to aspect direction
## Length = the distance from source rounded to the nearest cell width

try:
    import sys
    sys.path.append("C:\Program Files\QGIS 3.20.1\apps\Python39\Lib\site-packages\osgeo")
    ## Import all the necessary modules
    #import os, sys, osgeo.gdal, numpy, math, time, traceback
    #I need to find a way of exiting python without exiting QGIS; which sys.exit does!
    import os, osgeo.gdal, math, time, numpy
    from osgeo.gdalconst import *

    startTime = time.time()
    print(startTime)
```

```

##
# Define fan angle and fan increment
FanAngle = 10
FanIncrement = 10
RegressionLimiter = 50
##

OutputWorkspace = "G:\geoscience\GeoHazards\Projects\Launceston Geoguide\SouthLauncestonRevision\LandslideModelling\Slide Susceptibility\ModelWorkspace"

Workspace = "G:\geoscience\GeoHazards\Projects\Launceston Geoguide\SouthLauncestonRevision\LandslideModelling\Slide Susceptibility\ModelWorkspace"

print ("Workspace folder is " + Workspace)

lssrc = Workspace+"\\"+'lssrc_clean.tif'
dem = Workspace+"\\"+'dem.tif'
beta = Workspace+"\\"+'beta.tif'
aspect = Workspace+"\\"+'aspect.tif'
print ('file paths are ',dem, aspect, beta,lssrc)
#Set the working directory
os.chdir(Workspace)
#
# Register tif driver
driverTif = osgeo.gdal.GetDriverByName('GTiff')
driverTif.Register()
#
#Open each input raster as a dataset and test that they can be opened
sourceDS = osgeo.gdal.Open(lssrc)
demDS = osgeo.gdal.Open(dem)
aspectDS = osgeo.gdal.Open(aspect)
betaDS = osgeo.gdal.Open(beta)

if sourceDS is None:
    print ('abandoning process because cannot open source raster')
    quit()
if demDS is None:
    print ('abandoning process because cannot open dem raster')
    quit()
if aspectDS is None:
    print ('abandoning process because cannot open aspect raster')
    quit()
if betaDS is None:

```

```

    print ('abandoning process because cannot open beta raster')
    quit()
# Get georeferencing information from the source raster
sourceTransform = sourceDS.GetGeoTransform()
xOrigin = sourceTransform[0] # coordinate of top left X
yOrigin = sourceTransform[3] # coordinate of top left Y
pixelWidth = sourceTransform[1] # width of a pixel
pixelHeight = sourceTransform[5] # height of a pixel

# Test that the rasters have the same alignment? This must be true!
demTransform = demDS.GetGeoTransform()
aspectTransform = aspectDS.GetGeoTransform()
betaTransform = betaDS.GetGeoTransform()

##test
print (xOrigin, demTransform[0], aspectTransform[0], betaTransform[0])
print (yOrigin, demTransform[3], aspectTransform[3], betaTransform[3])
##
if xOrigin != demTransform[0] or xOrigin != aspectTransform[0] or xOrigin != betaTransform[0]:
    print ("abandoning process because rasters are not aligned along X axis")
    quit()
else: print ("rasters are aligned along Xaxis")
if yOrigin != demTransform[3] or yOrigin != aspectTransform[3] or yOrigin != betaTransform[3]:
    print ("abandoning process because rasters are not aligned along Y axis")
    quit()
else: print ("rasters are aligned along the Y axis")
#
    # Test that the pixels are square! This must be true
if pixelWidth != pixelHeight * -1:
    print ("abandoning process because pixels must be square! pixelWidth = " + str(pixelWidth) + " and pixelHeight
= " + str(pixelHeight * -1))
    quit()
else: print("pixels are square", pixelWidth, pixelHeight)
#
    # Create variable "MeanCellSize"
MeanCellSize = pixelWidth

# Get image size and create variables: rows and cols
rows = sourceDS.RasterYSize
cols = sourceDS.RasterXSize

```

```

print("Rows and columns",rows, cols)
##
# Test that the rasters are the same dimensions. This must be true!
if rows != demDS.RasterYSize or rows != aspectDS.RasterYSize or rows != betaDS.RasterYSize:
    print ('Abandoning process: image rows are not the same size')
    quit()
else: print ("image rows are the same size")
if cols != demDS.RasterXSize or cols != aspectDS.RasterXSize or cols != betaDS.RasterXSize:
    print ('Abandoning process: image columns are not the same size')
    quit()
else: print ("image columns are the same size")

# Get RasterBand from the input rasters (these should be single band "1-indexed")
sourceBand = sourceDS.GetRasterBand(1)
demBand = demDS.GetRasterBand(1)
aspectBand = aspectDS.GetRasterBand(1)
betaBand = betaDS.GetRasterBand(1)

# Determine the NoData value of the rasters
aspectNull = aspectBand.GetNoDataValue()
demNull = demBand.GetNoDataValue()
betaNull = betaBand.GetNoDataValue()

print ("null values = ", aspectNull, demNull,betaNull)
# Create array objects for inputs
sourceArray = sourceBand.ReadAsArray(0, 0, cols, rows).astype(numpy.float)
demArray = demBand.ReadAsArray(0, 0, cols, rows).astype(numpy.float)
aspectArray = aspectBand.ReadAsArray(0, 0, cols, rows).astype(numpy.float)
betaArray = betaBand.ReadAsArray(0, 0, cols, rows).astype(numpy.float)

#Create regression array: note that the order of rows and columns is different for numpy arrays, but is correct!
regressionArray = numpy.zeros((rows, cols), numpy.int)
#
# Now for the modelling aspect in the multidimensional array space!
# Loop through the array, rows and columns
for i in range(rows):
    print(i, " out of ", rows, " rows." )
    for j in range(cols):

```

```

# Find a valid source cell and read all the other input grid values at the cell address
if sourceArray[i,j] > 1:
    AspectSource = aspectArray[i,j]
    ElevSource = demArray[i,j]
    regressionArray[i,j] = 1 # Mark the source cell as an arbitrary value = 1

# Skip each null cell in aspect or elevation grids
if ElevSource != demNull or AspectSource != aspectNull or AspectSource != -1:

    #Undertake regression modelling by fanning +/- about the FanAngle in increments (FanIncrements).
    This is done to eliminate any holes that may occur in the regression
        for fan in range(0-FanAngle, FanAngle, FanIncrement):

# Calculate the regression direction, opposite to Aspect, ensuring it is between 0 and <360, and taking
into account the fanning
AspectUphill = AspectSource + fan
if AspectUphill < 0:
    AspectUphill = AspectUphill + 360
elif AspectUphill >= 360:
    AspectUphill = AspectUphill - 360

if AspectUphill >= 180:
    AspectUphill = AspectUphill - 180
else:
    AspectUphill = AspectUphill + 180
# Calculate incremental dx and dy to determine next runout cell address
dX = math.sin(math.radians(AspectUphill))
dY = math.cos(math.radians(AspectUphill)) * -1 #the "-1" is needed because the origin is top left
# Reset other variables
X = j
Y = i
Length = 0
elevBetaLine = ElevSource

# Create temporary variable to allow 'while loop' below to work
RegressionLoop = True
# Start stepping out on the regression path
while RegressionLoop == True:
    X = X + dX #increment x
    Y = Y + dY #increment y

```

```

RndX = int(round(X, 0)) # this allows the array coordinates to be accessed (they are integers)
RndY = int(round(Y, 0)) # this allows the array coordinates to be accessed (they are integers)

#If the cell X,Y goes beyond the grid limit; then skip to next fan angle increment
if RndX < 0 or RndY < 0 or RndX > cols -1 or RndY > rows -1:
    RegressionLoop = False
    break

else:
    # If elevation value is null skip to next fan angle increment
    if demArray[RndY,RndX]== demNull:
        RegressionLoop = False
        break

    # If beta value is null skip to next fan angle increment. Given that this is dependent on geology. It
    is considered useful having this error check here
    elif betaArray[RndY,RndX] == betaNull:
        RegressionLoop = False
        break

    else:
        # Calculate new elevation for Beta line. This is responding to changes of beta value as the path
        is followed
        elevDiffBeta = math.tan(math.radians(betaArray[RndY,RndX]))* MeanCellSize
        elevBetaLine = elevDiffBeta + elevBetaLine
        # Determine elevation of ground surface
        elevCell = demArray[RndY,RndX]
        # Increment the horizontal regression distance one cell
        Length = Length + MeanCellSize
        # Test that regression has not gone past the limit
        if Length > RegressionLimiter:
            break

        # Test whether to assign a value to the regression array or not
        #Check to see that
        if sourceArray[RndY,RndX] < 2: #This is not a potential source cell and could potentially write to it
            if regressionArray[RndY,RndX] == 0:
                regressionArray[RndY,RndX]= Length
            elif Length < regressionArray[RndY,RndX]:
                regressionArray[RndY,RndX]= Length

        # Test to determine if the Beta line is above or below the ground elevation to determine whether

```

to stay in the loop

```

        if elevBetaLine >= elevCell:
            RegressionLoop = False # Change the flag to allow escape to next regression line once the
following test is performed
            break
# Modelling is now completed! Write out the regression array.
# Create output raster as an unsigned 16bit integer (0 or positive values)

OutputTif = OutputWorkspace+"\\ "+ "regress.tif"
outputDS = driverTif.Create(OutputTif, cols, rows, 1, GDT_UInt16)
if outputDS is None:
    print ("Could not create dataset")
    quit()

outBand = outputDS.GetRasterBand(1)

    # Write the output data, syntax: array, xOffset, yOffset
outBand.WriteArray(regressionArray, 0, 0)

    # Flush data to disk, set the NoData value and calculate stats
outBand.FlushCache()
stats = outBand.GetStatistics(0, 1)

    # Georeference the output raster
outputDS.SetGeoTransform(sourceTransform)

    # Set the projection of the output raster
outputDS.SetProjection(sourceDS.GetProjection())

#Memory Management: set variables to none
sourceDS = None
aspectDS = None
demDS = None
betaDS = None
outputDS = None
print ("Regression modelling for map sheet is completed" )
#Final step is to calculate the processing time as all of the map sheets have been processed
endTime = time.time()
print ("The script took ' + str((endTime - startTime)/60) + ' minutes to complete')
except:
    print("something went wrong")

```

# APPENDIX 2

**RUNOUT SCRIPT (PYTHON 3.7)**

## Runout script (Python 3.7)

```
## Author: Colin Mazengarb
## The script is written in Python 3, and runs in the python console of QGIS 3.
## Successfully tested in QGIS 3.16
## Purpose of Script: A script that models the runout of deep seated landslides
## The script uses an alpha value derived from the geology vector layer to control.
# Runout is controlled by three limiters:
# 1) the Conefall method where runout is not so strongly controlled by local topography.
# 2) total travel distance:source distance ratio of 3.5 to prevent unreasonably long runouts (developed for the NW
Coast of Tasmania), particularly where thin basaltic units (potentially susceptible to failure) overlies non-susceptible
basement units that are deeply eroded.
# 3) An arbitrary length limiter as a further control on the runout to prevent unreasonable runouts
## All input raster file formats are to be tifs, of identical extent, cell size and fully aligned
## "lsrc" = the name of the source grid. This is created through a query of a slope grid; shall consist of source areas
= 1, non source areas = 0 or null
## "dem" = the name of the digital elevation model; any floating point values or null.
## "aspect" = the name of the aspect grid; positive floating point numbers ranging from 0 to <360, or -1 (horizontal).
## "alpha" = the name of the alpha angle grid.
## integers, 0 < alpha < 90
## "runout" = the name of the output runout grid; shall consist of positive integers representing distance (multiples
of the cell size) from the source cell. Where a runout cell is identified in more than one runout path, the lower value
will take precedence.
## A limit on the amount of runout can be performed as a subsequent query (using the map calculator in GIS if needed

## DEFINITION of some TEMPORARY VARIABLES
## X= coordinates of analysed cell
## dX= increment of length for each runout step
## Y= coordinates of analysed cell
## dY = increment of length for each runout step
## Length = the distance from source rounded to the nearest cell width
## FanAngle = (degrees) an angle of spread for the runout direction on either side of the source cell aspect
## FanIncrement = the number of runout trajectories on the FanAngle
## RunoutRatioLimiter = total travel distance:source distance ratio of 3.5 to prevent unreasonably long runouts (see
above)
## RunoutLimiter = an arbitrary length limiter as a further control on the runout to prevent unreasonable runouts
try:
    ## Import all the necessary modules
    import os, sys, osgeo.gdal, numpy, math, time, traceback
    from osgeo.gdalconst import *

    startTime = time.time()
    print ("started at " + str(startTime))
```

```

# Define fan angle and fan increment
FanAngle = 20
FanIncrement = 10

# Define runout limiter distance in metres. Runout restricted to 50m
RunoutLimiter = 50

#Define ratio of source distance to runout distance
RunoutRatioLimiter = 3.5

OutputWorkspace = "G:\geoscience\GeoHazards\Projects\Launceston Geoguide\SouthLauncestonRevision\LandslideModelling\Slide Susceptibility\ModelWorkspace"

Workspace = "G:\geoscience\GeoHazards\Projects\Launceston Geoguide\SouthLauncestonRevision\LandslideModelling\Slide Susceptibility\ModelWorkspace"
print ("Workspace folder is " + Workspace)

lssrc = Workspace+"\\"+'lssrc_clean.tif'
dem = Workspace+"\\"+'dem.tif'
alpha = Workspace+"\\"+'alpha.tif'
aspect = Workspace+"\\"+'aspect.tif'
print ('file paths are ', dem, aspect, alpha, lssrc)

#Set the working directory
os.chdir(Workspace)

# Register tif driver
driverTif = osgeo.gdal.GetDriverByName('GTiff')
driverTif.Register()

#Open each input raster as a dataset and test that they can be opened
sourceDS = osgeo.gdal.Open(lssrc)
demDS = osgeo.gdal.Open(dem)
aspectDS = osgeo.gdal.Open(aspect)
alphaDS = osgeo.gdal.Open(alpha)

if sourceDS is None:
    raise Exception('abandoning process because cannot open source raster')

elif demDS is None:
    raise Exception('abandoning process because cannot open dem raster')

```

elif aspectDS is None:

```
    raise Exception('abandoning process because cannot open aspect raster')
```

elif alphaDS is None:

```
    raise Exception('abandoning process because cannot open alpha raster')
```

else:

```
    # Get georeferencing information from the source raster
```

```
    sourceTransform = sourceDS.GetGeoTransform()
```

```
    xOrigin = sourceTransform[0] # coordinate of top left X
```

```
    yOrigin = sourceTransform[3] # coordinate of top left Y
```

```
    pixelWidth = sourceTransform[1] # width of a pixel
```

```
    pixelHeight = sourceTransform[5] # height of a pixel
```

```
    # Test that the rasters have the same alignment? This must be true!
```

```
    demTransform = demDS.GetGeoTransform()
```

```
    aspectTransform = aspectDS.GetGeoTransform()
```

```
    alphaTransform = alphaDS.GetGeoTransform()
```

```
if xOrigin != demTransform[0] or xOrigin != aspectTransform[0] or xOrigin != alphaTransform[0]:
```

```
    raise Exception('abandoning process because rasters are not aligned along X axis')
```

```
elif yOrigin != demTransform[3] or yOrigin != aspectTransform[3] or yOrigin != alphaTransform[3]:
```

```
    raise Exception('abandoning process because rasters are not aligned along Y axis')
```

```
    # Test that the pixels are square! This must be true
```

```
    elif pixelWidth != pixelHeight * -1:
```

```
        raise Exception('abandoning process because pixels must be square! pixelWidth = ' + str(pixelWidth) + ' and  
pixelHeight = ' + str(pixelHeight * -1))
```

```
    # Create variable "MeanCellSize"
```

```
    MeanCellSize = pixelWidth
```

```
    # Get image size and create variables: rows and cols
```

```
    rows = sourceDS.RasterYSize
```

```
    cols = sourceDS.RasterXSize
```

```

# Test that the rasters are the same dimensions. This must be true!
if rows != demDS.RasterYSize or rows != aspectDS.RasterYSize or rows != alphaDS.RasterYSize:
    raise Exception('Abandoning process: image rows are not the same size')

elif cols != demDS.RasterXSize or cols != aspectDS.RasterXSize or cols != alphaDS.RasterXSize:
    raise Exception('Abandoning process: image columns are not the same size')

#All happy with datasets, now final setup
# Get RasterBand from the input rasters (these should be single band "1-indexed")
sourceBand = sourceDS.GetRasterBand(1)
demBand = demDS.GetRasterBand(1)
aspectBand = aspectDS.GetRasterBand(1)
alphaBand = alphaDS.GetRasterBand(1)

# Determine the NoData value for each of the rasters
aspectNull = aspectBand.GetNoDataValue()
demNull = demBand.GetNoDataValue()
alphaNull = alphaBand.GetNoDataValue()
print('aspectNull = ',aspectNull)
print('demNull = ',demNull)
print('alphaNull = ',alphaNull)

# Create array objects for inputs
sourceArray = sourceBand.ReadAsArray(0, 0, cols, rows).astype(numpy.float)
demArray = demBand.ReadAsArray(0, 0, cols, rows).astype(numpy.float)
aspectArray = aspectBand.ReadAsArray(0, 0, cols, rows).astype(numpy.float)
alphaArray = alphaBand.ReadAsArray(0, 0, cols, rows).astype(numpy.float)

# Create a runout array:
# Note that the order of "rows and cols" in the statement below is correct despite being the opposite of the above
arrays. Apparently this is a numpy convention where Y is listed before X
runoutArray = numpy.zeros((rows, cols), numpy.int)

#Optional bit to calculate progress. Firstly calculate the number of source cells
TotalSourceCells = 0
for i in range(rows):
    for j in range(cols):
        if sourceArray[i,j] > 1:
            TotalSourceCells = TotalSourceCells + 1
print("There are ", TotalSourceCells, " source cells to process")

```

```

runoutcounter = 0 # Calculates the total number of runout cells regardless of whether they run over source areas
SourceCounter = 0 # Counter to calculate percentage of sources cells processed
maxrunoutcounter = 0 # Calculates the longest single runout
# Loop through the array
for i in range(rows):
#   print(i, " out of ", rows, " rows." )
    if SourceCounter % 100 == 0:
        print(i, SourceCounter, int((SourceCounter/TotalSourceCells) * 100))
    for j in range(cols):
        if sourceArray[i,j] > 1:
            SourceCounter = SourceCounter + 1
            runoutArray[i,j] = 1 #Mark any source cells just for QA purposes
            # Find valid source cells, and non-null dem, aspect,
            if aspectArray[i,j] != aspectNull and aspectArray[i,j] != -1 and demArray[i,j] != demNull and alphaArray[i,j]
            != alphaNull and alphaArray[i,j] != 0:
                ElevSource = demArray[i,j]
                alphaAngle = alphaArray[i,j]

            #Now do runout model for each part of the fan
            for FanAspect in range(0-FanAngle, FanAngle, FanIncrement):
                #Initialise variables: X, Y & L need to be reset for each increment of the fan
                X = j
                Y = i
                L = 0 # L is the horizontal distance variable
                SourceLength = 0 # length over a source area
                testline201 = 201
                # Calculate the runout direction with the fan component
                RunoutDirection = aspectArray[i,j] + FanAspect

                # Ensure that FanAspect is between 0 and < 360
                if RunoutDirection < 0:
                    RunoutDirection = RunoutDirection + 360
                elif RunoutDirection >= 360:
                    RunoutDirection = RunoutDirection - 360

                # Calculate dx and dy components from the vector (runout direction and one pixel distance)
                dX = math.sin(math.radians(RunoutDirection))
                dY = math.cos(math.radians(RunoutDirection))* -1 #this is needed because the origin is top left and
                would give the wrong direction if not used

```

```

# I am limiting the runout to five steps below (50m) as part of my testing. This can be increased later
for ro in range (0,5,1):
    X = X + dX #increment x
    Y = Y + dY #increment y
    L = L + MeanCellSize
    RndX = int(round(X,0)) # this allows the array coordinates to be accessed (they must be integers)
    RndY = int(round(Y,0)) # this allows the array coordinates to be accessed (they must be integers)

    # If the cell X,Y goes beyond the grid limit; then skip to next fan angle increment
    if RndX < 0 or RndY < 0 or RndX > cols - 1 or RndY > rows - 1:
        break #Next fan increment

    # If aspect or dems have null values then skip
    elif aspectArray[RndY,RndX] == aspectNull or demArray[RndY,RndX] == demNull:
        break #Next fan increment

    # Find the elevation value for cell address RndX,RndY
    ElevRunout = demArray[RndY, RndX]

    # Calculate the travel angle (A) by calculating elevation difference from source to runout cell / length
    A = math.degrees(math.atan((ElevSource - ElevRunout) / L))

    # Incrementing the source length variable if the runout is over source material
    if sourceArray[RndY, RndX] != 1:
        SourceLength = SourceLength + MeanCellSize
    # If the shadow angle limiter has been exceeded - escape out of loop
    if A < alphaAngle:
        break #Next fan increment
    # If the the total source limiter ratio is exceeded - escape out of loop
    elif SourceLength > 0 and L / SourceLength > RunoutRatioLimiter:
        break #Next fan increment
    # If the length has gone over a nominated value - escape out of loop
    elif L > RunoutLimiter:
        break #Next fan increment
    else: # A valid runout cell has been identified
        runoutcounter = runoutcounter + 1
        # Mark source cells as 1
        if sourceArray[RndY,RndX] > 1:
            runoutArray[RndY, RndX] = 1
        # Mark valid non source cells as L

```

```

        elif runoutArray[RndY, RndX]== 0 or L < runoutArray[RndY, RndX]:
            runoutArray[RndY,RndX] = L
        continue

# Modelling is now completed! Write out the runout array.
# Create output raster as an unsigned 16bit integer (0 or positive values)
OutputTif = OutputWorkspace+"\\runout.tif"
outputDS = driverTif.Create(OutputTif, cols, rows, 1, GDT_UInt16)
if outputDS is None:
    raise Exception('Could not create output tif')

outBand = outputDS.GetRasterBand(1)

# Write the output data, syntax: array, xOffset, yOffset
outBand.WriteArray(runoutArray, 0, 0)

# Flush data to disk, set the NoData value and calculate stats
outBand.FlushCache()
stats = outBand.GetStatistics(0, 1)

# Georeference the output raster
outputDS.SetGeoTransform(sourceTransform)

# Set the projection of the output raster
outputDS.SetProjection(sourceDS.GetProjection())

#Memory Management: set variables to none
sourceDS = None
aspectDS = None
demDS = None
alphaDS = None
outputDS = None

print ("Runout modelling for map sheet is completed")

print ('Total runout cells =', runoutcounter)

#Final step is to calculate the processing time
endTime = time.time()

```

```
print ('The script took ' + str(round((endTime - startTime)/60, 2)) + ' minutes to complete')
```

```
except:
```

```
    print ("something went wrong")
```



Tasmanian  
Government

**Mineral Resources Tasmania**

PO Box 56 Rosny Park

Tasmania Australia 7018

Ph: +61 3 6165 4800

[info@mrt.tas.gov.au](mailto:info@mrt.tas.gov.au)   [www.mrt.tas.gov.au](http://www.mrt.tas.gov.au)