

SOFTWARE METAPAPER

Benthic Photo Survey: Software for Geotagging, Depth-tagging, and Classifying Photos from Survey Data and Producing Shapefiles for Habitat Mapping in GIS

Jared Kibebe¹

¹ Leigh Marine Laboratory, Institute of Marine Science, University of Auckland, New Zealand
jkibebe@gmail.com

Photo survey techniques are common for resource management, ecological research, and ground truthing for remote sensing but current data processing methods are cumbersome and inefficient. The Benthic Photo Survey (BPS) software described here was created to simplify the data processing and management tasks associated with photo surveys of underwater habitats. BPS is free and open source software written in Python with a QT graphical user interface. BPS takes a GPS log and jpeg images acquired by a diver or drop camera and assigns the GPS position to each photo based on time-stamps (i.e. geotagging). Depth and temperature can be assigned in a similar fashion (i.e. depth-tagging) using log files from an inexpensive consumer grade depth / temperature logger that can be attached to the camera. BPS provides the user with a simple interface to assign quantitative habitat and substrate classifications to each photo. Location, depth, temperature, habitat, and substrate data are all stored with the jpeg metadata in Exchangeable image file format (Exif). BPS can then export all of these data in a spatially explicit point shapefile format for use in GIS. BPS greatly reduces the time and skill required to turn photos into usable data thereby making photo survey methods more efficient and cost effective. BPS can also be used, as is, for other photo sampling techniques in terrestrial and aquatic environments and the open source code base offers numerous opportunities for expansion and customization.

Keywords: Remote Sensing; Habitat Mapping; Marine Ecology; GIS; Ground Truth; Validation; Photography; Image Processing; Geotagging; Software; Benthic Survey

Funding statement: BPS was developed as part of a PhD project funded by the Auckland Council with additional support from DigitalGlobe Foundation.

(1) Overview

Introduction

Geotagged photos provide an efficient means of gathering spatially explicit information for environmental monitoring, resource management [17], ecological studies [6, 9, 15], and habitat mapping methods where it is necessary to ground truth habitat types or detect changes [1, 5]. Such photo survey methods have advantages over more traditional observational techniques [4] as field workers do not need to be trained in interpretation and the photos provide a permanent visual record that can be revisited and reinterpreted in the future. Geotagged photos can then be classified visually (e.g. habitat, species presence/absence, etc.), exported as a spatially explicit point shapefile suitable for analysis in GIS, and compared with remote sensing data [13, 14]. However, the conversion of classification attributes and field data (photos and GPS logs) into shapefile format is a cumbersome and labor intensive process involving multiple pieces of software and manual data entry that is prone to error. Comparisons of photographic

and observational methods have cited the time and technical expertise required for this process as major barriers to use of photographic methods in GIS [12, 13].

In the marine environment there are inherent difficulties with carrying out benthic photo surveys. Seafloor photos must be taken in close proximity to the bottom due to the attenuation of light in water and to capture images of sufficiently high resolution. Habitat mapping requires that photos are geotagged, but GPS receivers will not function below the water's surface so using cameras with integrated GPS is not feasible. Photos can be captured underwater via drop camera or in the water on snorkel or SCUBA [14] but GPS records must come from a receiver on the boat (when used with a drop camera) or on a diver-towed float. Depth measurements can aid in the mapping of benthic habitats [7, 16] but previously described methods do not provide for depth integration with photo survey results.

Benthic Photo Survey (BPS) was created to streamline the production of shapefiles from photo surveys of the

seafloor and integrate depth measurements into the process. BPS provides a simple graphical user interface for batch geotagging, manual habitat and substrate attribution of each photo, and exporting this information to a shapefile. In contrast with earlier methods BPS is also designed to incorporate depth and temperature data from an inexpensive depth logger [11] attached to the camera. BPS requires a GPS log, a set of photos, and (optionally) a depth log to be simultaneously collected in the field. With these data as input (**Fig. 1**), BPS can write location and depth information into the Exchangeable image file format (Exif) portion of each jpeg photo as a batch process. The user can then visually assign habitat and substrate categories to each photo, which are also written in the Exif portion of each image. Habitat and substrate categories can be defined by the user through a preferences menu for application in any environment. Once all attributes have been assigned, a shapefile can be exported from BPS where the location of each photo is represented as a point feature. Each of these features is attributed with all available data (see the BPS website [2] for more detail including a full list of feature attributes). The exported shapefile can be used with most GIS software (QGIS [10] for example) for further analysis. A step by step BPS tutorial for use with the included test data (images, GPS log, and depth log) is available on the web [3].

Implementation and architecture

BPS is primarily intended to be used via the graphical user interface (GUI). Most users will not require any understanding of the software beyond that necessary to install and run it. Users who intend to use BPS to process their photo transect data should direct their attention to the BPS website and documentation [2, 3]. However, the following may be of interest to those who wish to customize the application, extend its capabilities, or simply understand what BPS does in greater detail.

The core functionality of BPS is contained within the following modules:

- bps_gui
- depth_temp_log_io
- gps_log_io
- photo_tagging
- bps_export

The BPS GUI was designed with QT Designer 4.8.6 (<http://qt-project.org/doc/qt-4.8/designer-manual.html>). The resulting user interface (UI) files were then used to generate the Python components of the user interface: ui_bps.py, ui_pref_help.py, and ui_preferences.py. These components, in turn, are imported by the bps_gui module. This module implements the functioning of the GUI and calls all the other subordinate modules into use.

The depth_temp_log_io module reads depth and temperature data from the Sensus Ultra dive data logger (<https://reefnet.ca/products/sensus/>), and writes it into a SQLite database. It also contains functionality for nearest timestamp lookups in the database and several unit conversions including time zone shifts.

The gps_log_io module reads GPS log data (position and timestamp), in gpx or NMEA text format, into the SQLite database. It also contains methods for converting coordinates and performing lookups in the database.

The photo_tagging module defines objects to assist in handling images and writing information into the Exif portion of jpeg files. It also provides the “Depth Plot” functionality accessible via the GUI and described in the BPS documentation [3].

Quality control

Testing for BPS was carried out by following the steps of the tutorial [3] using the included test data and manually verifying that the output was as expected.

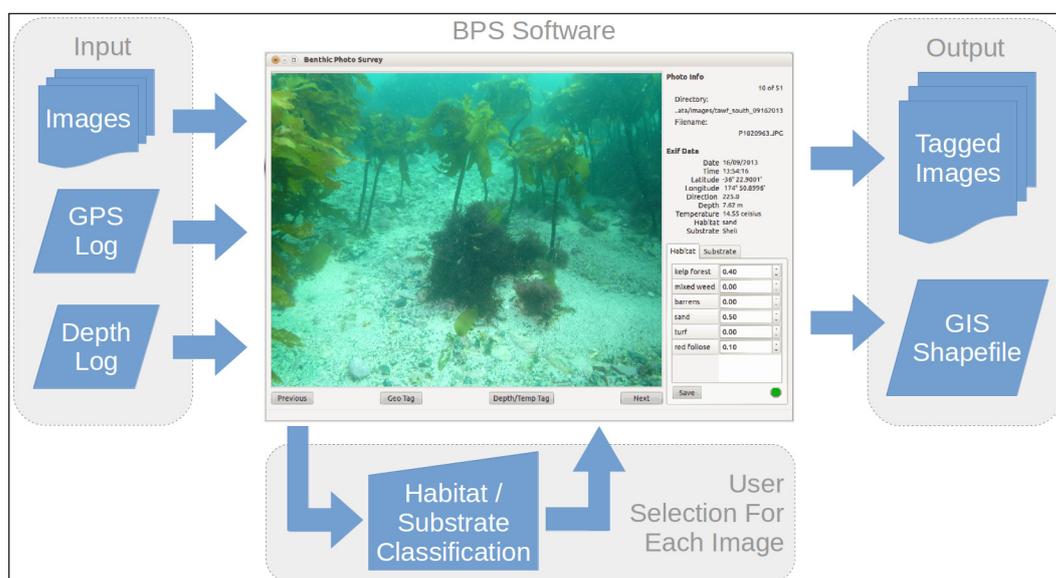


Figure 1: Graphical representation of BPS workflow. Geotagging and depth tagging are batch processes. Habitat and substrate classification require user input for each image.

This testing procedure was conducted on Ubuntu 14.04 and Windows 7.

(2) Availability

Operating system

BPS is primarily intended for use on Ubuntu or Windows operating systems. The full installation of BPS and its dependencies to run from source on Ubuntu is a simple procedure. This full installation, while possible on Windows, is more complicated. Therefore, a stand-alone Windows executable file is available. Instructions for Ubuntu full installation and Windows stand-alone installation are provided on the BPS website [2].

BPS was developed and tested on Ubuntu 14.04 and Windows 7. As a Python application, it should be possible to run it on any operating system that can support Python and the other dependencies listed below but no testing has been carried out to date.

Programming language

Python 2.7. BPS has not been tested for compatibility with Python 3.

Additional system requirements

BPS has been successfully operated on virtual machines (Windows and Ubuntu) with as little as 1GB of RAM and a single Intel i5 core. While testing has not been conducted at a lower specification, this configuration may be considered the minimum requirement.

Support

For assistance with BPS, please contact the author via email (jkibele@gmail.com). Bug reports and feature requests can be posted to the BPS issue tracker (https://github.com/jkibele/benthic_photo_survey/issues).

Dependencies

When running the Windows binary version of BPS, it is not necessary to install any of these dependencies. Compiled versions are included as part of the Windows binary download. When running BPS from source, the following dependencies are required:

- Python 2.7.
- Scipy 0.13.3 Used for interpolating between measured depths.
- Matplotlib 1.3.1 Used for plotting depth profiles.
- PyQt4 4.8.6 Used to develop the graphical user interface.
- GDAL 1.10 Used for writing shapefiles and handling projections.
- Pyexiv2 0.3.2 Used to read and write Exif data.
- Pynmea 0.6.0 Used to read GPS NMEA log files.
- Slugify 0.0.1 Used to clean up user input.

Installation of these dependencies is simple on the Ubuntu 14.04 operating system and instructions are provided on the installation section of the BPS website [2]. Running from source is possible on other operating systems (e.g. Windows and OSX) but the installation and configuration

of dependencies can be tricky and that process is beyond the scope of this document.

List of contributors

Jared Kibele

Software location

Archive

Name: Zenodo.org

Persistent identifier: <http://doi.org/10.5281/zenodo.33028>

Licence: New BSD

Publisher: Jared Kibele

Date published: 02/11/2015 v1.0.6

Code repository

Name: GitHub

Identifier: https://github.com/jkibele/benthic_photo_survey

Licence: New BSD

Date published: 23/08/2012 initial commit on bitbucket.org. Moved to GitHub in Nov 2014. v1.0.6 published 02/11/2015

Language

English.

(3) Reuse potential

BPS was developed to facilitate the processing of field data (underwater photos, GPS logs, and depth logs) into shapefiles for use as ground truth in the derivation of subtidal habitat maps from multispectral satellite imagery. However, it is likely to be suitable for many other uses in both aquatic and terrestrial environments. This section will detail some of the ways in which BPS could be used in the current version of the code and then discuss additional uses that would require modifications to the code.

In the current version of the code, BPS is useful for producing a point shapefile from photos and a GPS log. In the simplest use case, BPS operates as a geotagging application. BPS also offers the option of adding user adjustable categorical attributes (habitat and substrate) and depth to each photo and corresponding shapefile location, giving it utility beyond that of a simple geotagging application. The following are some potential uses:

- In the absence of satellite or aerial imagery, subtidal habitat/bathymetry maps could be generated by simply collecting a large number of evenly spaced photos and interpolating the BPS shapefile.
- Ground truth data could be generated for terrestrial habitat mapping by walking with a GPS and photographing the scene at ground level.
- In a citizen science context, field data (photos and GPS/depth logs) could be collected by volunteers with minimal training and processed using BPS by researchers who may not even be on site.
- Visually observable traits other than habitat and substrate could also be mapped. For instance, one could use the BPS preferences menu to define the substrate categories as numerical bins (e.g. 0–5, 5–10,

10–15, . . ., etc.), attribute each photo according to the abundance of a given species visible in the photo, and use the resulting shapefile to produce a map of the abundance of this species. With depth-tagging, the shapefile could also be used to investigate the relationship between the density of a species and depth.

Regardless of the specific task, BPS writes location and all other available data (depth, temperature, habitat, substrate, etc.) to the Exif metadata portion of each jpeg as well as to the generated shapefile. This makes the jpeg files permanent records on their own with potential uses outside of BPS. For instance, a collection of BPS attributed photos could easily be cataloged in a PostGIS database and made available online as a historical archive. With additions over time, such an archive could serve as a record of change in habitat or species distribution.

The free and open source Python code for BPS could also be modified to alter or expand its capabilities. For instance, additional sensors that produce a log of time-stamped readings could easily be coded for. If one were interested in mapping variations in pH, new code could be written to parse the log files and select the appropriate readings to match the photo time-stamps. The code for dealing with the depth/temperature log files could serve as a template for any time-stamped log file. Modifications could also be made to the BPS user interface to add functionality. Useful additions might include some of the functionality available in Coral Point Count [8] such as area measurement and random point overlays.

Competing Interests

The author declares that they have no competing interests.

Acknowledgements

Thanks to Dr. Nick Shears and Dr. Richard Taylor who have supervised the author's PhD and provided valuable input.

References

1. **Andréfouët, S**, et al. 2003 (Nov.) Multi-site evaluation of IKONOS data for classification of tropical coral reef environments. *Remote Sensing of Environment*, 88(1–2): 128–143. DOI: <http://dx.doi.org/10.1016/j.rse.2003.04.005>
2. **Benthic Photo Survey** 2014 Available at http://jkibele.github.io/benthic_photo_survey/ (Accessed: 2015-01-15).
3. **Benthic Photo Survey Documentation** 2014 Available at <http://benthic-photo-survey.readthedocs.org/en/latest/> (Accessed: 2015-01-15).
4. **Coyer, J A**, et al. 2012 *The underwater catalog: a guide to methods in underwater research*. Shoals Marine Laboratory.
5. **Dekker, A G**, et al. 2011 Intercomparison of shallow water bathymetry, hydro-optics, and benthos mapping techniques in Australian and Caribbean coastal environments. *Limnology and Oceanography: Methods*, 9: 396–425. DOI: <http://dx.doi.org/10.4319/lom.2011.9.396>
6. **Done, T**, et al. 2007 (Dec.) Decadal changes in turbid-water coral communities at Pandora Reef: loss of resilience or too soon to tell? *Coral Reefs*, 26(4): 789–805. DOI: <http://dx.doi.org/10.1007/s00338-007-0265-3>
7. **Eugenio, F**, et al. 2015 (Jul.) High-Resolution Maps of Bathymetry and Benthic Habitats in Shallow-Water Environments Using Multispectral Remote Sensing Imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 53(7): 3539–3549. DOI: <http://dx.doi.org/10.1109/TGRS.2014.2377300>
8. **Kohler, K E** and **Gill, S M** 2006 Coral Point Count with Excel extensions (CPCe): A Visual Basic program for the determination of coral and substrate coverage using random point count methodology. *Computers & Geosciences*, 32(9): 1259–1269. DOI: <http://dx.doi.org/10.1016/j.cageo.2005.11.009>
9. **Kutser, T**, et al. 2007 (Dec.) Photo-library method for mapping seagrass biomass. *Estuarine, Coastal and Shelf Science*, 75(4): 559–563. DOI: <http://dx.doi.org/10.1016/j.ecss.2007.05.043>
10. **Quantum GIS Development Team** 2011 *Development Team, 2012. Quantum GIS Geographic Information System. Open Source Geospatial Foundation Project.*
11. **ReefNet Inc. | Sensus Ultra**. Available at <https://reefnet.ca/products/sensus/> (Accessed: 2015-09-22).
12. **Roelfsema, C M**, et al. 2004 Evaluating benthic survey techniques for validating maps of coral reefs derived from remotely sensed images. *International Coral Reef Symposium Proceedings*.
13. **Roelfsema, C** and **Phinn, S** 2010 (Apr.) Integrating field data with high spatial resolution multispectral satellite imagery for calibration and validation of coral reef benthic community maps. *Journal of Applied Remote Sensing*, 4(1): 043527–043527. DOI: <http://dx.doi.org/10.1117/1.3430107>
14. **Roelfsema, C** and **Phinn, S R** 2009 A Manual for Conducting Georeferenced Photo Transects Surveys to Assess the Benthos of Coral Reef and Seagrass Habitats version 3.0. Centre for Remote Sensing and Spatial Information Science, The University of Queensland, Brisbane, Australia.
15. **Ruhl, H A** 2007 Abundance and size distribution dynamics of abyssal epibenthic megafauna in the northeast Pacific. *Ecology*, 88(5): 1250–1262. DOI: <http://dx.doi.org/10.1890/06-0890>
16. **Sagawa, T**, et al. 2010 Using bottom surface reflectance to map coastal marine areas: a new application method for Lyzenga's model. *International Journal of Remote Sensing*, 31(12): 3051–3064. DOI: <http://dx.doi.org/10.1080/01431160903154341>
17. **Schlacher, T A** et al. 2010 High-resolution seabed imagery as a tool for biodiversity conservation planning on continental margins. *Marine Ecology*, 31(1): 200–221. DOI: <http://dx.doi.org/10.1111/j.1439-0485.2009.00286.x>

How to cite this article: Kibele, J 2016 Benthic Photo Survey: Software for Geotagging, Depth-tagging, and Classifying Photos from Survey Data and Producing Shapefiles for Habitat Mapping in GIS. *Journal of Open Research Software*, 4: e10, DOI: <http://dx.doi.org/10.5334/jors.104>

Submitted: 08 November 2015 **Accepted:** 08 March 2016 **Published:** 23 March 2016

Copyright: © 2016 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.

]u[*Journal of Open Research Software* is a peer-reviewed open access journal published by Ubiquity Press

OPEN ACCESS 