

Objectives		Description	Requirements	Processing environment	Media type
Front end development	Input	Input data can be any kind of printed book with range maps The book series of V. Tshikolovets serves as examples here	Books	Library	Analog books
	Scan	A device for efficiently scanning pages from books or the services of a digitization lab are needed	Scan or digitization device Digitization services	Library	Analog book pages
	Storage	Pages are preferably stored in tiff format for low compression bias Storage space is needed	Storage capacity	Hard drive	Tiff files
	Preprocessing	Double pages must be clipped to single pages Rectification is needed to correct the scanned images Boarder clipping should be applied to reduce storage demands	Implementation	Scan tailor C++ functions	Tiff files
	Object detection	Page elements must be identified. Elements include text, maps, images, tables Further processing will use detected maps Adding other elements to the processing pipeline is planned for the future	Implementation	openCV, fastai	Tiff files
	Pixel classification	Different methods will be implemented, e.g. black and white thresholding, selection by colour range, k-means clustering and mean shift	Implementation	scipy, sklearn	Tiff files
	Georeferencing	Ground control points (GCPs) and projection details must be provided by the user for batch processing of the input files	Implementation	gdal functions	Text file with GCPs and projection details
	Postprocessing	Contour fitting with alpha shapes	Implementation	numpy	Classified tiff files
	Output	Multiple output formats are desirable for flexible further processing	Storage capacity	Hard drive	Geo tiff files Geo json files Shape files
	Validation	Comparision with manually digitized results and/or original data (e.g. percentage overlap)	Implementation	raster, sp	Shape files