Different components of RootSkel

Component	File name	Description
1	Main script	Main file that calls all subfunctions via the internal *MATLAB* callback
	Root_image_GUI.m	
2	Log files	Log files documenting all changes including dates so changes can be undone and future developers can build upon the existing version
	Log.txt	Since the last version
	Old_Versions_log.txt	All previous versions
	CurrentVersion.txt	A shorter version of previous log files including bug fixes
3	Functions	Folder containing the 18 subfunctions
	▷ var_saver.m	★ Creates a variable *varnames* which contains the names of the relevant variables (*skelmatR, skelmatR_simp, max_curv_point, savename*); it then pulls them from the base workspace and lets the user save them in a .mat file. ★ *skelmatR* or *skelmatR_simp* include the skeleton of the root (their x and y coordinates), *max_curv_point* includes the user's input for the possible turning point or an empty set, *savename* includes the name of the image (date and hour) and the number of the roots which is used for names of figures, first column in .csv file and default of *var_saver.m*
	▷ var_loader.m	★ Allows the user to load the .mat files including the relevant objects from the workspace ★ Contains the enabling of appropriate angle calculation buttons; buttons are disabled to avoid bugs and errors (eg angle computation on nothing should not work)
	▷ skel_crop.m	★ Contains the optional free hand cropping of the skeleton
	▷ skel_clean.m	★ Loops on optional additional cleaning, ie bigger and bigger objects are removed, until user is satisfied
	▷ savename_crea.m	★ Saves the label of the root or root number the user chooses in order to keep track of which root is analysed ★ Combines the label with the name of the file and saves it as a folder where the variables (see above) would go
	▷ **root_skel.m**	★ Takes results from *image_process.m* ★ Applied more fine-tuned filtering ★ Applies more cleaning steps ★ Tries to makes sure that the root tip is in the skeleton ★ Combines approach 1 and 2 ★ Returns the skeleton
	▷ point_get.m	★ Asks the user for points as long as she does not provide the required number (defined as a number of points between minimum and maximum) ★ The user's input is stored in the strings *srcx* and *srcy* are strings with the name of the variable that will receive the data in the base workspace; they tell *assignin* in which variable in the caller to store the data
	▷ point_choose.m	★ Collects the necessary points from the user: 5 points close to the tip, 5 - 10 evenly spaced points on the desired root starting with the tip, the tip of the root ★ Each step can be redone
	▷ image_zoom.m	★ Inverts the image ★ Lets the user zoom in (and zoom out via right click)
	▷ **image_process.m**	★ Extracts the cropped image ★ Extracts the colours from the sample pixels and averages it with a certain neighbourhood (3x3) ★ Takes a brightness range, an average of the three filters used ★ Approach 1: Colour separation filtering • based on RGB values of points • gray scales image ★ Approach 2: Brightness filtering (intensity-based approach) • enhances brightness • eliminates too bright spots
	▷ image_crop.m	★ Optional free hand cropping
	▷ image_choose.m	★ Allows the user to choose an image ★ Modifies the image using various filter to help the user discern the root
	▷ **getAngle.m**	★ Takes the skeleton as input ★ Computes the curvature and angle of the root tip
	▷ force_tip.m	★ Prompts user to create an open polygon between the edge of the current skeleton and the tip ★ In order to make sure that the tip of the root is definitely included in the skeleton
	▷ final_prep.m	★ Extracts only the tip of the root and the respective x and y values which are passed on to *getAngle.m* ★ User can choose to select the turning point, ie point with highest local curvature, which can serve as another verification of the computed turning point; it does not have to be exactly on the root as the point in the skeleton that is closest to the chosen point is used
	▷ fig_saver.m	★ Saves the relevant objects upon clicking different buttons
	▷ fig_loader.m	★ Loads respective figures
	▷ angle_file.m	★ Creates a file *root_angles.csv* or *user_assisted.csv* depending on *user_flag* and prints the label of the root and the angle; this file can be appended for consecutive angle calculations of the same root in other images