

# TreeSnatcher Plus



## Tutorial 1 Graphical User Interface Overview of all activities

The work view displays an image of a phylogenetic tree. The image is overlaid with a representation of the tree topology. Blue nodes are branching positions, green nodes are taxa, the green lines are logical links (branches) between them. The lower window shows a Newick format representation of the tree depicted. The Newick expression can be copied to the clipboard or exported to a text file. You may type in taxon names or use the numbers the program assigns to each tip.

Path to current file

The screenshot shows the TreeSnatcherPlus application window. At the top, the title bar indicates the path to the current file: `/Users/thomas/Desktop/TreeSnatcherPlus/snapshot/snap24.ser`. The main workspace displays a phylogenetic tree with a semi-transparent image overlaid. The tree has blue nodes for branching positions and green nodes for taxa. A red dashed box highlights a portion of the tree, with a red line indicating a branch length of 105. A key at the bottom left of the tree lists: VAM only (if known), Some ECM, and Many ECM. Below the tree, a Newick format representation is shown: `((((4:155, (2:154, 3:155):0):0, (5:150, (7:127, (10:107, (11:105, (12:79, (13:54, (6:28, 14:29):25):23):24):0, (8:103, 9:105):0):23):24):24):0, (15:177, (17:152, (((27:129, 37:128):0, ((40:55, (22:29, (30:31, 31:29):0):24):24, (28:54, (38:30, (29:30, 39:32):0):25):23):50):23, ((16:129, (18:128, 19:130):0):0, ((23:126, ((21:77, (25:55, (26:30, 32:30):25):24):24, (20:103, 24:103):0):24):0, (33:129, (34:56, ...`. The interface includes several toolbars: File Operations (Open Image, Open Snapshot, Save Image, Save Snapshot, New Image, Save Newick), Selection Tools (Whole Image, Grab Motif, Box Selection, Line Selection, Move Node, Drag Branch), Drawing Tools (Pencil, Rubber, Black Line, White Line, Curve, Fill, Pipette, Stencil), and various processing tools like Greyscale Conversion, Global Threshold, Local Threshold, Sharpen, Despeckle, Minimum, Brightness, Flood Foreground, Extract Foreground, Locate Nodes, Calculate Branches, and Get Scale Bar Length. A status bar at the bottom provides information about the selected object: `Branch from Tip at (288, 151) to Inner Node at (184, 151) is 106 pixels long, has calculated length 105, is almost straight, co...`. On the right side, there are panels for Blend Ratio, Preprocessing Tools, Recognition Tools, Tree Type, and Settings.

8x Magnifier

Load/Save/New image  
Load/Save current state  
Save Newick expression

Select an image portion  
Move/manipulate a node  
Drag a branch  
Grab a portion as a motif

Modify the image  
Paste a grabbed motif  
Manipulate colors  
Adjust pencil and rubber size

Adjust the blend ratio between original image and modified image

Prepare the original image  
Separate foreground and background  
Thin the image foreground

After preparation, mark the foreground that is the tree  
Place the nodes of the tree  
Determine its branches  
Scale the tree

TreeSnatcher Plus treats rectangular and freeform trees differently

Functions that simplify the preparation of the original image

Show/Hide frame with Newick representation  
Undo functionality

Displays the color under the mouse position in the modified image

A status bar informs about the currently selected object and activities

What follows is a list of the available activities in TreeSnatcher Plus. An activity can be issued from the pop-up menu which becomes visible after pressing the right mouse button, or from the application menu, or from the left and right tool bars. Some functionality can be found exclusively in the application menu.

All image operations act on the work-in-progress image and not on the original image. To avoid confusion, you might want to adjust the **Blend Ratio** accordingly.

The source image serves as a reference: It contains the species names that you might want to type in, shows the exact position of line intersections, the original branches etc. If it contains a scale bar, you can use it to scale the tree.

All operations are illustrated in the tutorials.

## File Operations

New Image	Create an image on your own. For instance, you might want to sketch your own tree
Open Image	Open an image in either of the formats PNG, JPG/ JPEG, GIF
Save Image	Save the current image as PNG. You may compose different layers of visual information.
Open Snapshot	Restore a snapshot of a progress of work.
Save Snapshot	Save a snapshot of the current progress of work. A snapshot contains all the data necessary to continue working on the tree and its topology at a later date.
Save Newick	Save resulting Newick expression.

# Selection Tools

Nearly all image operations require you to preselect the area they act on. If nothing happens after you have issued an activity, check whether an image portion is selected.

**Box Selection** A rectangular selection.

**Line Selection** A straight line selection. You first select a line of known length in the image, i.e. a scale bar, then you issue the activity **Get Scale Bar Length** to scale the tree.

**Move Node** Select this activity if you need to move nodes on the image foreground. This can be necessary if they are placed too far from a line intersection.

**Drag Branch** Allows you to define an artificial branch between two nodes. This is only sensible if you assign the branch length manually and/or create trees on your own. Why? If the program has not placed a logical branch between two nodes this means that it has not found a foreground path between them. If you are sure that there should be a branch, check whether the path contains no gap. Additionally, make sure that the nodes are placed at line crossings.

You should not normally require to use this activity at all.

**Whole Image** Make a box selection that encompasses the whole image. You should use it for the image operations **Binarization** and **Thinning**.

**Grab Motif** Grabs the image portion within the selection box as the stencil motif.

# Drawing Tools

All drawing operations act on the work-in-progress image, not on the source image. In order to observe the effects on the work-in-progress image, you should adjust the blend ratio.

Using the drawing tools, your primary goal is to prepare the image for the placement of nodes and branches. For this, you need to separate image foreground (the tree!) and background. Leave as few colors in the image as possible. The higher the contrast between dark foreground and bright background, the better.

Prior to drawing, you need to convert the image to grayscale, perform thresholding on it, then thin it. Use the drawing tools together with **Flood Foreground** to mark the foreground that represents the tree.

Please study the tutorials for a better overview.

- |                   |  |
|-------------------|--|
| <b>Pencil</b>     | Freehand drawing with shade black. Use it for instance to fill gaps in the foreground. When <b>Thin Freehand Drawings</b> is activated, your freehand drawings are thinned on the fly. Thinning is necessary for the correct placement of nodes and branches. When <b>Flood User Drawings</b> is activated, the program automatically floods your freehand, line and curve drawings that you appended to a formerly flooded foreground region. |
| <b>Rubber</b>     | Freehand drawing with shade white. Use it for instance to erase foreground pixels.   |
| <b>Black Line</b> | Line drawing with shade black. If <b>Mask User Drawings</b> is activated, the program surrounds your line and curve drawings with borders in the background shade (white). This is particularly useful if you need to draw out branches that are overlaid with text.   |
| <b>White Line</b> | Line drawing with shade white.   |
| <b>Curve</b>      | Quadratic curve drawing with shade black. You may first define start point and end point, then drag the handle for the placement of the third point.   |

Pipette	Allows you to pick the RGB color of the pixel under the mouse position as the fill color. The selected color is displayed in the left tool bar. The precise RGB color composition at the mouse coordinate is shown in the bottom left-hand corner of the TreeSnatcher Plus window. Use this activity in conjunction with <b>Fill</b> .
Fill	Fill the image portion within the selection box with the color you have chosen using <b>Pipette</b> .
Stencil	Paste the motif you selected using <b>Grab Motif</b> at the mouse position.
Pencil Width Rubber Width	Adjust the width for the drawing tools to a value between 1 and 30. <b>Thin Freehand Drawings</b> only works with pencil width 1.
Fill Dialog	<p>A dialog which allows you to perform various color range manipulations on the image. You may replace one color or a selection of colors with another color, or you keep one color or a selection of colors and replace the remaining colors with another color.</p> <p>There are situations in which you can use the fill dialog to reduce the number of colors in the image.</p>
Fill Selection Black	Fill the image portion within the box selection with shade black.
Fill Selection White	Fill the image portion within the box selection with shade white.
Clear all	Fill the whole image with shade white.

# Image Operations

- Trim** Fills the area outside the selection box with shade white.
- Crop** Selects the area inside the selection box as a new image, discards the rest.  
This activity cannot be undone.
- Double Size** Scales the whole image by a factor of 2.  
This operation can be useful when the foreground structures in the image are fuzzy, i.e. when parallel branches overlap, lines have too many pixels in common etc.  
This activity cannot be undone.
- Half Size** Scales the whole image by a factor of 0.5.  
This activity cannot be undone.

# Preprocessing Tools

A grayscale image in TreeSnatcher Plus is a RGB image in which each component in a color triplet is assigned the same value from the range 0..255. TreeSnatcher Plus automatically removes an alpha channel in the source image.

## Invert

Inverts the area within the selection box in a color image.  
You might use this operation if the image foreground is bright and background dark.

## Histogram Stretch

Performs a histogram stretch with saturation on a grayscale image. The result is misleading when used on a color image.  
This operation tries to improve the contrast of the image.  
The implementation ignores the lower and upper 0.5% of the shades.

The new shade is determined in the following way:  
$$\text{New shade} = 255 * (\text{old shade} - \text{lowest shade}) / (\text{highest shade} - \text{lowest shade})$$

## Color Quantization

Tries to reduce the amount of colors in the image. You may select the desired number of colors. As the result is often not intuitive, you should experiment with it.

## Grayscale Conversion

Converts the color image to a grayscale image. You may adjust the contribution of each of the color channels to the new shade. Informs you whether or not the contrast remains subjectively stable. This is the case if the factors roughly add up to 1.0.

## Global Threshold

This is the recommended option to create a binary image from a grayscale image.  
Lets you specify a threshold shade between 0 and 255.  
The shades greater than or equal to the threshold become black, the others white.

For the automatic node placement to work, the whole image section that contains the tree must be binarized.

## Local Threshold

Performs local binarization of an grayscale image portion using a separated Gauss filter. It moves a small window line-by-line over the image and locally computes a suitable threshold value for the image portion within the window. The result is not always what you want. In such cases you should use **Global Threshold**.

## Smooth

Smooths a color image portion using a Gauss Filter. You may adjust the size of the kernel between 1 and 5.

## Sharpen

Increases the apparent sharpness of a color image portion using the Unsharp Masking technique. You may select a kernel width between 1 and 25 and the amplification factor between 0.1-fold and 5-fold.

This operation is efficient if you need to clean blurred line structures; the thresholding becomes easier after sharpening.

## Despeckle

Performs median filtering, a rank operation, on the colors of an image portion. You may specify a box size between 1 and 10.

Use this operation to remove small holes in foreground structures or straighten contours. Technically speaking, this filter reduces noise in the image while preserving edges in many cases.

## Minimum

Performs minimum filtering, a rank operation, on the colors of an image portion. You may specify a box size between 1 and 10.

Use this operation to enhance dark values in the image by increasing its area.

## Brightness

Brightens or darkens a portion of an image. You may specify the factor with which a color gets multiplied and the offset that is added to it.

## Thin

Skeletonizes the foreground of a binary image. The goal of the operation is to shrink the size of all foreground structures as much as possible. A line width of one pixel is the ideal. A skeletonized foreground is the prerequisite for the correct placement of nodes and branches. This is why the option **Thin Freehand Drawings** is there.

# Recognition Tools

The recognition tools are used after the original color image has been preprocessed, i.e. trimmed to size, converted to grayscale, thresholded, thinned, modified with the drawing tools etc.

The drawing tools and the recognition tools can also be used in turn. If the node placement does not fulfil your expectations, you may modify the image and restart the node placement. If not all branches have been found it is likely that there is a gap somewhere on the foreground between a pair of nodes that define the affected branch. You can use the pencil to fill it. The activity **Flood Foreground** can reveal the position of the gap if there are foreground portions which are not filled but should be.

## Flood Foreground

This activity signals the program where the tree is. The tree is a coherent portion of the foreground of an image that has been thresholded (binarized) and thinned (skeletonized). The foreground is black and the background is white.

Starting from a foreground position that you selected via right mouse click, all foreground pixels reachable from this position are colored in blue. The „flooding“ stops at gaps in the foreground.

You add or subtract foreground portions from/to the already flooded region by drawing and erasing pixels. It can be of help to activate **Mask User Drawings**, **Thin Freehand Drawings** and **Flood User Drawings**.

The position from where the flooding operation has been initiated can be permanently marked with a cross. For this, activate **Mark TreeFlood Seed**.

## Extract Foreground

This is a convenience operation. Discards anything but the flooded foreground.

## Place Nodes

Uses a heuristic to determine the positions of line intersections in the tree, then places inner nodes there. It also determines the positions of line endings in the image, then places outer nodes there. The nodes are visualized as green and blue dots, respectively. You may add, move and remove nodes.

Line intersections are those locations in a phylogenetic tree at which a speciation event occurred. An end of a line usually represents the position of a species/taxon/gene.

## Calculate Branches

Employs a path finding approach to discover the foreground paths between two neighboring nodes. If the path is not closed the program will not form a branch. When the program has found a path between two nodes it measures its length in pixels and forms a new branch object. This logical branch is visualized as a straight line between the nodes.

This automated branch placement is the normal modus operandi in TreeSnatcher Plus.

Manual branch placement via **Drag Branch** is discouraged. If you use it anyway, you must specify the branch length manually. You can use manual branch lengths and calculated branch lengths simultaneously when you select **Use Mixed Lengths**.

If you can do without branch lengths in the resulting Newick representation, you can use **Drag Branch** without risk. You might want to deselect **Newick Expression with Lengths** then.

Name Taxon

Allows the user to type in a name for a species/taxon/gene. As a default, the branches are assigned numbers in ascending order.

Use Node As Origin

This option determines the inner node from which the Newick expression for the tree is constructed. This node is not necessarily identical to the tree root.

Input User Branch Length

Allows you to type in a length for the currently selected branch.

Use Length 1.0 to Scale Tree

Assigns the value 1.0 to the length of the currently selected branch and rescales the tree with the new branch length.

Use Length to Scale Tree

Lets you first specify a length for the selected branch, then rescales the tree with the new branch length.

Get Scale Bar Length

Obtains the length in pixels of the current line selection, then lets you type in the length of this line in a custom metric (the tree metric). You may mark a line of known length in the tree or a scale bar depicted in the image.

Freeform

Informs the program that the tree has no special topology. The branch length measured in pixels is identical with the length of the foreground path between the pair of nodes that define the branch.

## Rectangular

Informs the program that the tree is rectangular. This means that every branch in the tree is segmented into „horizontal“ and „vertical“ portions. Only the horizontal portions account for the length of a branch. The program is able to identify the different branch segments if they are sufficiently long and the bends in the lines are sharp enough.

Use the **Show Branch Length Composition** view to inspect whether the program has identified all branch segments correctly. Segments colored in red are length relevant, such colored in black are ignored.

If you are not satisfied with the segmentation of a branch, use the drawing tools to straighten the lines and accentuate the bend.

## Remove Node

Removes the currently selected node. If the topology has been calculated at least one time, it is recalculated.

## Remove Nodes in Selection

Removes all nodes within the selection box. If the topology has been calculated at least one time, it is recalculated.

## Remove Branch

First removes the currently selected branch, then recalculates the topology.

## Remove Branches in Selection

First removes all branches within the selection box, then recalculates the topology.

## Toggle Newick View

Displays or hides the Newick expression window.

## Use Elaborate Path Finding

The program aims at finding a foreground path between two nodes, starting at either of the nodes. It moves from pixel to pixel until it has reached or is in the vicinity of another node. It then builds a branch which is defined through the two nodes and which has the length of the Path measured in pixels. Afterward this path can be divided into straight segments with their individual length if the tree type **rectangular** is selected.

The program knows two strategies for the determination of branches:

If **Use Elaborate Path Finding** is deselected, the algorithm expects the inner nodes to lie at an intersection. It starts at either of the nodes and moves from pixel to pixel until it reaches another node. This approach is fast, but it comes at a cost: Fuzzily placed nodes are ignored, which will lead to wrong branches.

If **Use Elaborate Path Finding** is selected, the algorithm tries to be more accurate while being more tolerant: It does not expect inner nodes to lie exactly at an intersection but seeks to lengthen the path to the node that is nearest. For this, it searches the neighborhood for another node whenever it can „move“ in more than one direction. You may either type in the radius of this look-ahead area, or let the program suggest it. For this, use **Suggest Look-ahead distance**.

Rule of thumb:

**Use Elaborate Path Finding** when the foreground structures in the prepared image are clear and the paths are longer than a few pixels.

If not all branches were identified correctly, you should experiment with different look-ahead distances. If the result is still not convincing, you should modify the image with the drawing tools and restart **Calculate Branches**.

# Settings

## Magnifier Settings

**Show Source Image** shows a blend between the original image and the current state of work in the magnifier. If it is deselected, only the current state of work is shown. If **Show Source Image** is activated, **Highlight Foreground** colors the foreground pixels in a color you may select from a color dialog. Coloring the foreground can be useful for drawing operations as it clearly exposes the differences between the original and e.g. the skeletonized image. The **Blend Ratio** needs to be adjusted adequately.

## Use Fancy Cursors

If activated, TreeSnatcher Plus uses its own mouse cursors in the context of different activities. Otherwise, it uses the system crosshair cursor and the hand cursor.

## Mark TreeFlood Seed

Places a blue cross at the position from where the last **Flood Foreground** operation was issued. This seed position decides about which part of the foreground will be filled in blue color or rather what portion of the foreground is the tree. You set it by right click on a foreground pixel and then choosing **Flood Foreground** from the pop-up menu.

## Thin Freehand Drawing

Thins whatever you draw with the pencil. This is sensible as the foreground needs to be thinned anyway before the placement of nodes can take place.

## Mask User Drawings

Surrounds whatever you draw in black with a white border. This can be helpful e.g. when you retouch fuzzy lines in the image that shall be linked to the tree, or when you draw completely new structures.

## Flood User Drawings

Automatically issues a **Flood Foreground** operation at the last seed position. This integrates anything you append to an already existing flooded foreground portion into it.

## Draw Branch Lengths

Displays or hides the branch lengths. The number that is displayed as branch length depends on which length model you use and which tree type you have selected.

## Draw Taxon Names

Displays or hides the taxon/species/gene names.

## Small Nodes and Branches

If activated, the program uses smaller dots to represent nodes and thinner lines to represent branches. If the smallest distance between a pair of nodes is below 10 pixels, this option is set automatically.

## Highlight Branches Without User Length

If **Use Mixed Lengths** or **Use User Defined Lengths** is activated, the application colors the branches in orange for which the user has not yet specified a custom length but has already assigned a custom length to at least one other branch.

## Highlight Tips Without User Name

The application colors the tips in pink for which the user has not yet typed in a name.

## Newick Expression with Lengths

You can decide whether the Newick expression shall contain branch lengths.

## Use User Assigned Lengths

If selected, the program uses and displays only the branch lengths you assigned to them. This means it ignores completely the calculated lengths.

## Use Calculated Lengths

If selected, the program uses and displays the branch lengths it has calculated for the branches. If the tree type **rectangular** is chosen, the length is the sum of the length relevant branch segments. Otherwise it is the full length of the foreground path between the nodes.

## Use Mixed Lengths

A combination of user assigned and calculated lengths. If you specified a custom branch length, the program uses it, otherwise it uses the calculated length.

## Shaded Newick String

For your convenience, you can use different shades for the species names and the branch lengths in the Newick expression.

## Show All Branches

This view displays all nodes and all logical branches.

## Check Linked Branches

This view only displays the nodes. When you place the mouse over a node, all paths that emerge from this node are highlighted in yellow.

Use this option to check whether **Calculate Branches** has followed all paths between two nodes correctly. If a branch is missing, the program has not found the path between the respective nodes. Find out what has gone wrong, modify the image using the drawing tools and restart **Calculate Branches**.

## Show Branch Length Composition

This view shows how TreeSnatcher Plus has divided each branch into segments. Length relevant branch segments are red, other segments are black. If you have not selected **rectangular** as tree type, the whole path is colored red.

The fuzzier the image of the tree is, the more likely it is that the program cannot identify the branch segments. For this, it must decide which lines are straight, where the bends are and whether there is an orientation that all outer branches have in common.

If you are discontent with the result, you ought to straighten the respective branch segments and accentuate the bend in the line using the drawing tools.

The accuracy of branch length measurement depends on some factors: Skeletonization, node placement, detection of bends in a line, division into branch segments. Moreover, as the smallest distance that the program can measure in the image is one pixel, it can never be as exact as the program that created the original tree.