Detecting Tree Rings Using Classic Image Processing Filters

Jacob Karon
York University
4700 Keele Street, Toronto, Ontario
M3J 1P3
jaykaron@my.yorku.ca

Abstract

This paper reimplements a recent method of automatically detecting tree rings by Fabijańska et al. *(2017) [3]. The method takes an image gradient based and edge detection approach applying common steps like preprocessing, filtering, etc. Afterwards other classic types of filters are applied and their results are compared.*

1. Introduction

Identifying and measuring tree rings can reveal a wealth of information regarding the environment in which the ring was formed. The width of the ring is connected to many factors such as temperature and moisture. This, coupled with the fact that the rings can be accurately dated, makes dendroclimatology relevant to a wide array of fields like archeology and ecology. In short dendroclimatology can be used to make historical models of past climates.

However, the process of measuring tree rings is a very slow process when done by hand and requires a trained professional. While there does exist software to aid the process (such as CDendro and CooRecorder), they still require a professional to manually label each specimen, A fully automated process would speed up the process immensely, allowing a greater number of trees to be analyzed.

2. Prior Work

Computer assisted tree ring detection started being developed in the 1990s and research in the field has continued ever since. One of the earliest techniques proposed by Sheppard and Graumlich [5] used only a one dimensional representation of the intensity signal to estimate tree rings. As the field developed, the methods became more complex in order to increase accuracy and handle a larger number of species with different internal structure. When dealing with incomplete tree rings (such as those in the dataset used) the standard method has become to detect edges in the gradient map and resolve the edges into tree rings. Other techniques have been developed for cases where the entire cross section of the tree is present, such as active contours [4] and generalized Hough transforms [1].

In general, most of the techniques mentioned so far only work if there is some clear contrast between the ring and it surrounding area. However, this is not true for all types of wood. In ring-porous wood the rings are mostly defined by a ring of large pores that correspond to the early growth. Without a contiguous boundary demarcating the ring, it is notoriously hard to detect the rings in ring-porous using the standard methods. In recent years convolutional neural networks have been applied specifically to ring-porous species with great results [2].

The purpose of this paper is to reimplement the system of Fabijańska *et al*. [3], a recent example of the standard gradient method, and to experiment with different filtering methods to improve results. Specifically filtering with non-isotropic filters in order to take advantage of the pre-known orientation of the rings.

3. Dataset

The dataset being used is the same one as [3]. A collection of 215 samples from 12 species, where 4 of the species are conifers, 2 are ring-porous and 6 are diffuse-porous. The samples were extracted from live trees using an increment borer and are 0.5 cm in diameter. The cores were then cut and scanned with the rings perpendicular to the length of the image. The images were saved as PNG files of either 600 or 1200 dpi.

The dataset came with 10 samples with the rings marked by a professional dendrochronologist. Initially, there were plans to manually label more of the dataset, but due to the difficulties of manually labeling (the need for expertise and the time investment), this idea was abandoned. All evaluations were done using the profesionally marked samples. Using the marked samples the sensitivity and precision of the different methods can be calculated and compared.

4. Method

The following is a description of my implementation of the method described by Fabijańska *et al*. with minor tweaks mentioned where relevant. Following that, there will be a description of the alternate filtering techniques that were tested.

For images of every step of the process please view this slideshow hosted on Google Drive (link).

4.1. Preprocessing

The preprocessing stage simply consists of transforming the RGB image into HSV and using only the V channel, representing the brightness or vividness of the pixels, under the assumption that the V channel has the highest signal-to-noise ratio. This assumption is based on an earlier work by Cerdal *et al*. [1]. However, in that work only one species of tree was analyzed and while using only the V channel gives better results overall, on some samples it leads to worse results. It is not clear that on a species level processing only the V channel will always be the optimal option.

The preprocessing was done using built-in Matlab functions like `rgb2hsv` and `rgb2gray`.

4.2. Filtering

Fabijańska *et al*. proposed a very basic filtering method in order to remove small artefacts. They proposed a small averaging filter in the shape of a circle where the radius is dependent on the resolution of the image.

There is a small inconsistency in the source paper regarding the radius. They state that the radius R should be 3 pixels for images of 600 dpi and 5 pixels for those of 1200 dpi, but in their formula for the kernel R denotes the square root of the radius as shown in (1).

$$h(i,j) = \begin{cases} 1, & \text{if } i^2 + j^2 < R \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

After brief testing using the given values as the square root of the radius gives better results. Alternate filtering techniques will be discussed at the end of the method section.

4.3. Gradient Map

A gradient map is constructed from the filtered image using an enlarged version of the Sobel operator (even though the Sobel operator is not defined for any size other than 3×3 [6]). Increasing the kernel size has a dampening effect on the resulting gradient map making it less affected by small local changes in the gradient which hopefully correspond to noise and not to actual rings. Formula (2) was used to create Sobel operators of larger sizes.

$$h(i,j) = \frac{-i}{i^2 + j^2} \tag{2}$$

The size of kernel is left as an open parameter, 7×7 gives the best results when using the circular averaging filter, various sizes were better when using other filtering methods. Only the horizontal component of the operator was used as the rings are expected to be nearly horizontal and any vertical gradient is most likely noise.

4.4. Thresholding

The gradient map is processed column by column in order to produce the binary image. The local maxima of the column are found, the median peak height is calculated and maxima higher than the median are added to the binary image. The Matlab function `findpeaks` is used to find the local maxima.

4.5. Edge Linking

From the binary map discrete edges must be resolved. First the connected components of the binary map are computed. Then components are iterated through in order of their size, largest to smallest. For each component a straight line is fitted across the image that best matches the component. The line is then thickened (using `imdilate`). The intersection of the thickened line and the binary image is computed in an effort to match disconnected components of the same ring.

If the number of points in the intersection passes a certain threshold (another free parameter, 70% of the image width gives good results) then we assume it has detected a ring. Another line is fit to the intersection and is saved as the location of a ring. After a component has been processed it is set to zero so a component can only be considered once.

Matlab function `bwconncomp` is used to determine the connected components and `polyfit` is used to create the line of best fit.

5. Alternate Filtering Methods

As noted earlier the filtering method described by Fabijańska *et al*., while it does a fine job at mitigating the intensity of small perturbances, seems a little too conservative. Since the rings are expected to be nearly horizontal and cross the entire image, little information would be lost by a more aggressive filtering technique along the horizontal axis. The results for the filters are discussed in the next section.

5.1. Elliptical

The simplest modification would be to extend the original filter formula, that allows only for circles, to allow different radii for the x and y axis, resulting in an ellipse. In-

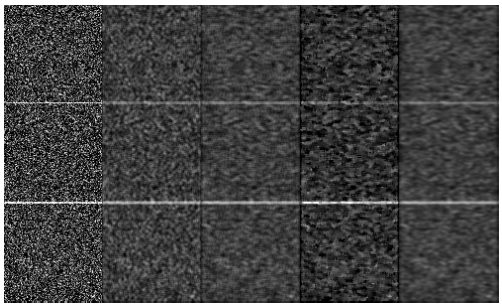

Figure 1. Side-by-side comparison of filtering methods. From left to right: original, circular, elliptical, median, gaussian

creasing the size of only the x axis preserves or even accentuates the horizontal lines while more actively removing noise that has little horizontal scope. Extending formula (1) for ellipses results in (3) where R_x and R_y are the radii of the two axes.

$$h(i,j) = \begin{cases} 1, & \text{if } i^2/{R_x}^2 + j^2/{R_y}^2 < 1 \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

5.2. Gaussian

A similar alternative to elliptical filters is a 2D Gaussian filter with different sigma values for the x and y axis. The benefit of Gaussian distributions over the elliptical filter is that pixels closer to the center are weighted more, retaining more localized structure. 2D Gaussian kernels can also be decomposed into two 1D distributions and applied separately, saving on computation time compared to the other filters for kernels of similar size.

5.3. Median

The final type of filter that was tried was a rectangular median filter. Median filters are know to be good at removing small amounts of noise while preserving edges. The median filter was implemented using Matlab's `medfilt2` function.

Figure **??** shows a side-by-side comparison of the filtering methods applied to an image of horizontal lines with Gaussian noise added. The alternate methods do a fine job of protecting the edges, especially when the edge is more than 1 pixel thick.

6. Results

Results were calculated based off the 10 marked samples. If the detected rings were within 3 pixels of the marked point they count as True Positives (TP), otherwise they are False Positives (FP). Non-detected rings are counted as False Negatives (FN). The statistics being measured are the sensitivity (SEN), precision (PREC), and the Dice coefficient (DICE).

Method	SEN	PREC	DICE	AVG
Fabijańska	0.6715	0.8582	0.7348	0.7548
Gaussian	0.6815	0.8093	0.7290	0.7400
Ellipse	0.6653	0.8509	0.7270	0.7477
Median	0.6647	0.7701	0.7009	0.7119

Table 1. Comparing results. Mean over all wood samples.

Method	SEN	PREC	DICE	AVG
Fabijańska	0.1935	0.3158	0.2400	0.2498
Gaussian	0.3226	0.3333	0.3279	0.3279
Ellipse	0.2258	0.3043	0.2593	0.2631
Median	0.2903	0.4500	0.3529	0.3644

Table 2. Comparing results. For ring-porous samples.

$$\text{SEN} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

$$\text{PREC} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

$$\text{DICE} = \frac{2\ \text{TP}}{2\ \text{TP} + \text{FP} + \text{FN}} \tag{6}$$

Oftentimes, something that increases the results for one of the measures will decrease results in others making it partially subjective as to which parameters are the best. Also, the different free parameters interact with one another making it hard to pin down the optimal choices. In an effort to find a base line I tried to find parameters that would lead to the highest average of all the measures across all the samples.

The parameters that maximized the average result following the method proposed by Fabijańska *et al*. were a Sobel operator of size 7 and an edge threshold of 70% of the image width. The results for the individual measures are in Table 1.

These results are slightly lower than the ones reported in [3]. This discrepancy is probably due to imperfect parameters, slightly different implementations and the difference in the dataset they were run on. However, it should serve well enough to compare against other filtering methods.

6.1. Comparing Methods

Unfortunately, while the other filters achieved similar results, none were any better than the circular filter across all the samples. Even though in theory the stronger filters should not interfere with horizontal lines, in reality things are not so clean and the stronger filters do more harm than good.

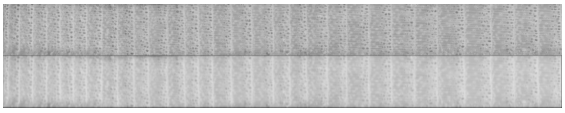

Figure 2. Top: The original ring-porous specimen. Bottom: Specimen after median filtering.

The one case which did see improvement from the other filters was on the ring-porous samples. In the case the circular filter performed the worst of all four with the Gaussian and median filters returning significantly better results (See Table 2).

In this case the heavy horizontal blurring actually helped, because ring-porous wood does not have clearly demarcated lines at ring boundaries but instead larger pores, the blurring of these pores made it easier for the boundaries to be detected (Figure 2).

7. Conclusion

While most of the alterations I wanted to make were ineffective, I learnt that conservative approaches are often the most effective when dealing with sensitive edges in noisy data sets.

In the future, more work should be done on the edge linking and thresholding parts of the process, as it has become apparent that the filter type is not particularly important.

References

[1] M. Cerda, N. Hitschfeld-Kahler, and D. Mery. Robust tree-ring detection. In *Pacific-Rim Symposium on Image and Video Technology*, pages 575–585. Springer, 2007.

[2] A. Fabijańska and M. Danek. Deepdendro–a tree rings detector based on a deep convolutional neural network. *Computers and Electronics in Agriculture*, 150:353–363, 2018.

[3] A. Fabijańska, M. Danek, J. Barniak, and A. Piórkowski. Towards automatic tree rings detection in images of scanned wood samples. *Computers and Electronics in Agriculture*, 140:279–289, 2017.

[4] P. Kennel, P. Borianne, and G. Subsol. An automated method for tree-ring delineation based on active contours guided by dt-cwt complex coefficients in photographic images: Application to abies alba wood slice images. *Computers and Electronics in Agriculture*, 118:204–214, 2015.

[5] P. Sheppard and L. Graumlich. A reflected-light video imaging system for tree-ring analysis of conifers. In *Tree rings, environment and humanity. Radiocarbon 1996. Proceedings of the International conference*, pages 17–21, 1994.

[6] I. Sobel. History and definition of the sobel operator. *Retrieved from the World Wide Web*, 2014.