Scalable max-tree and alpha-tree algorithm for high resolution, multispectral, and extreme dynamic range images

You, Jiwoo; Wilkinson, M.H.F.; Trager, Scott university of groningen

XXX Canary Islands Winter School of Astrophysics: Big Data Analysis in Astronomy (4th-10th Nov, 2018)

A Fast, Memory Efficient Alpha Tree Algorithm using Flooding and Tree Size Estimation

Jiwoo You[1,2] Scott C. Trager[1] Michael H. F. Wilkinson[2]
j.you@rug.nl
[1]Kapteyn Astronomical Institute, University of Groningen [2]Bernoulli Institute, University of Groningen

Introduction

Partition Tree (Max tree, Alpha tree)

- Tree data structures used in morphological image filtering
- Connected morphological filters are very useful for faint object detection, as shown by [1] and [2]
- Alpha tree is useful in analysis of satellite or planetary images [3]

A Novel Fast, Memory Efficient Alpha Tree Algorithm

- The first Alpha tree flooding algorithm
- The first study to accurately estimate the partition tree size to increase memory efficiency

Method

Alpha Tree Flooding Algorithm

- Algorithm design motivated from [2] and [4] with some modifications for α–trees

Alpha Tree Size Estimation (α–TSE)

- The tree size can be easily estimated from pixel dissimilarity histogram (*dhist*)

$$TSE(D) = Ne^{-\pi D} \qquad D = \frac{\sqrt{\sum_{e \in E} dhist[e]^2 - |E|}}{|E| - 1}$$

N: Image size
E: Set of neighbouring pixel pairs

- *D* is a root mean squared deviation between *dhist* and flat histogram
- We found that the tree size is an exponential decay function of *D*

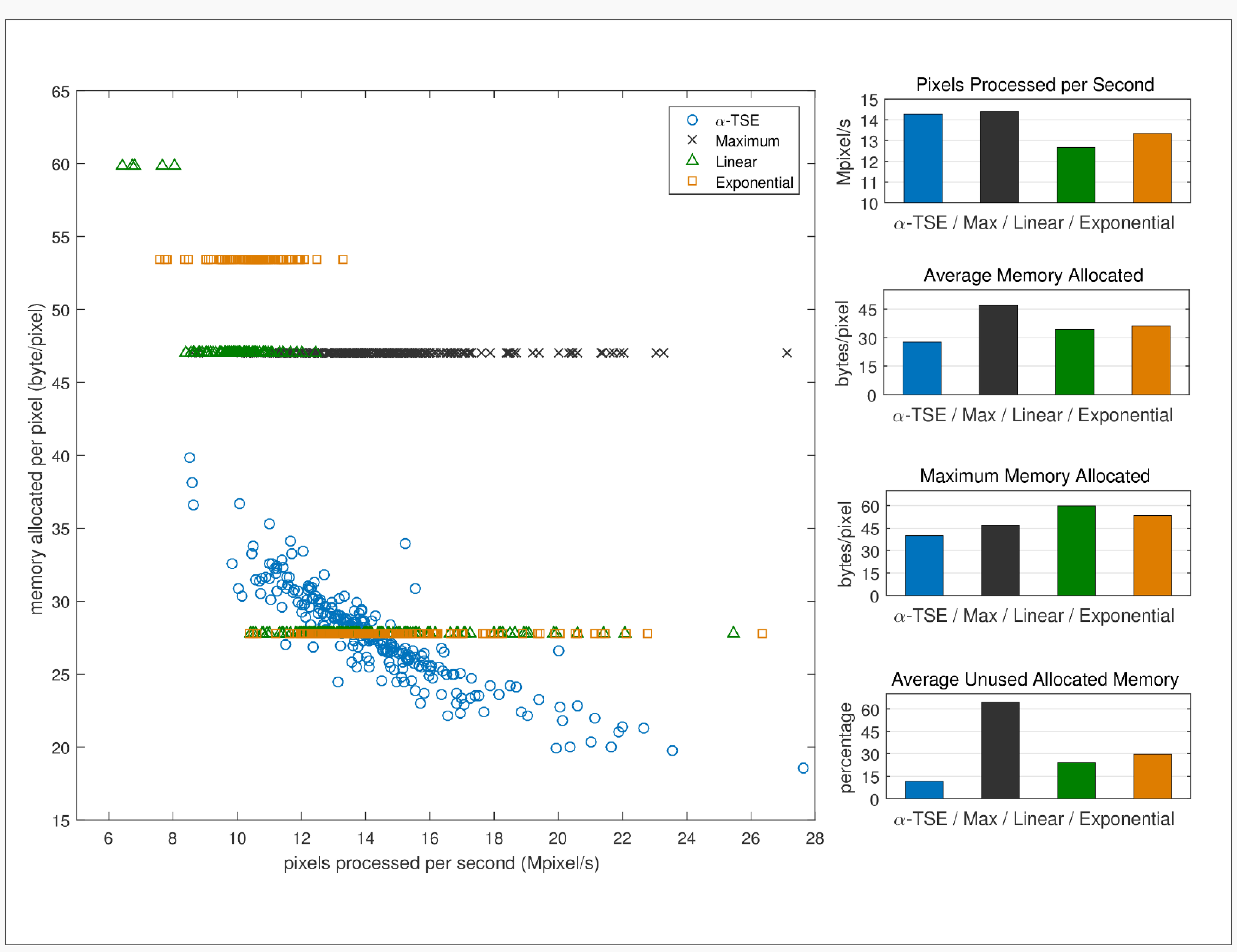
Results

The Test Dataset

- Manually collected 254 low dynamic range (8-bit) optical images
- Experiments conducted in both colour and grey-scale
- Results on grey-scale images are shown here

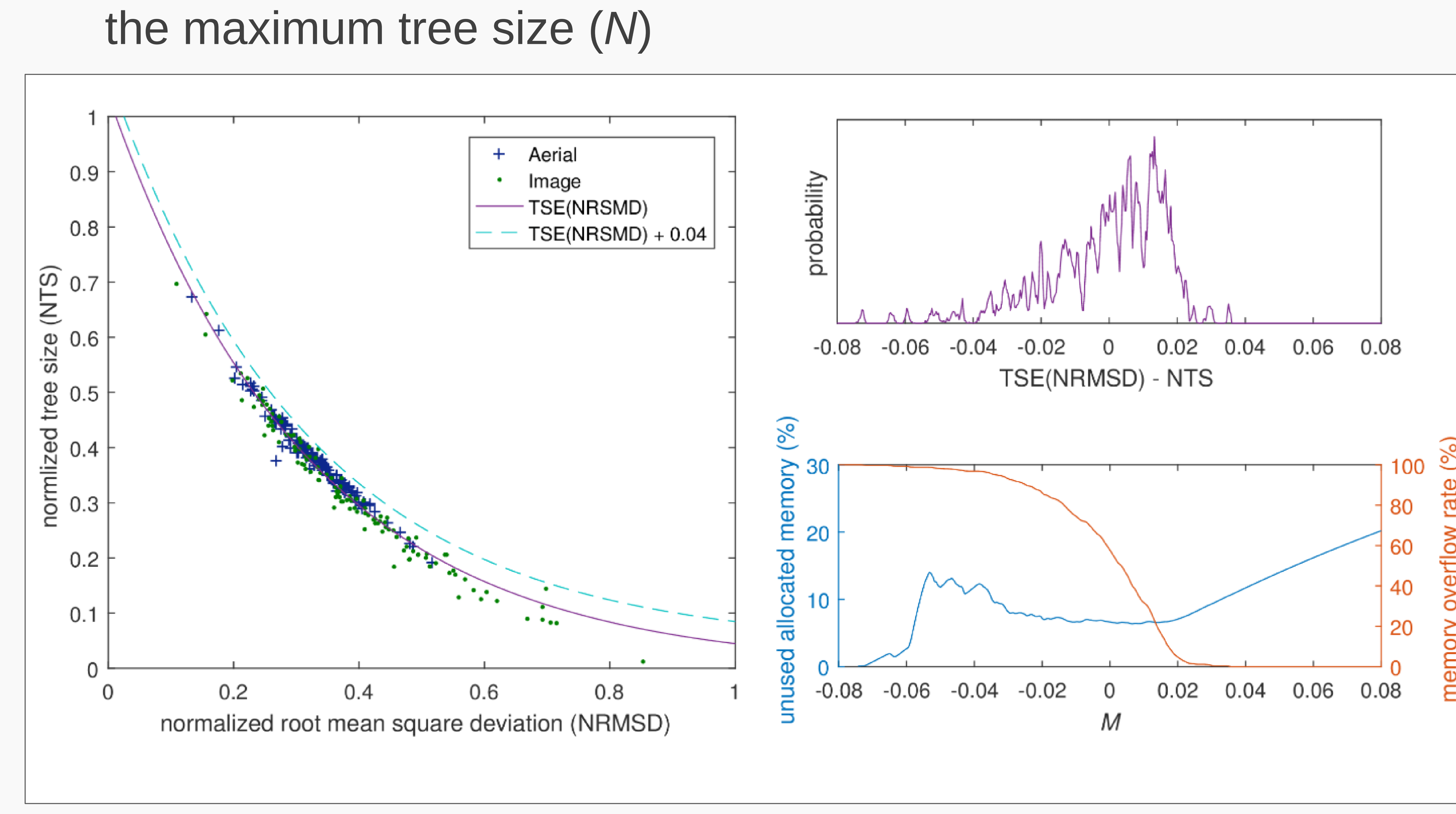
The α–TSE Performance

- The α–TSE reduced average memory usage by 41%
- Computation increase of α–TSE was only 0.3% (14.3 Mpix/s)
- The α–TSE performed better than other dynamic memory reallocation schemes
- Execution speed and memory usage in α–TSE were anti-correlated – Execution speed can be predicted using α–TSE



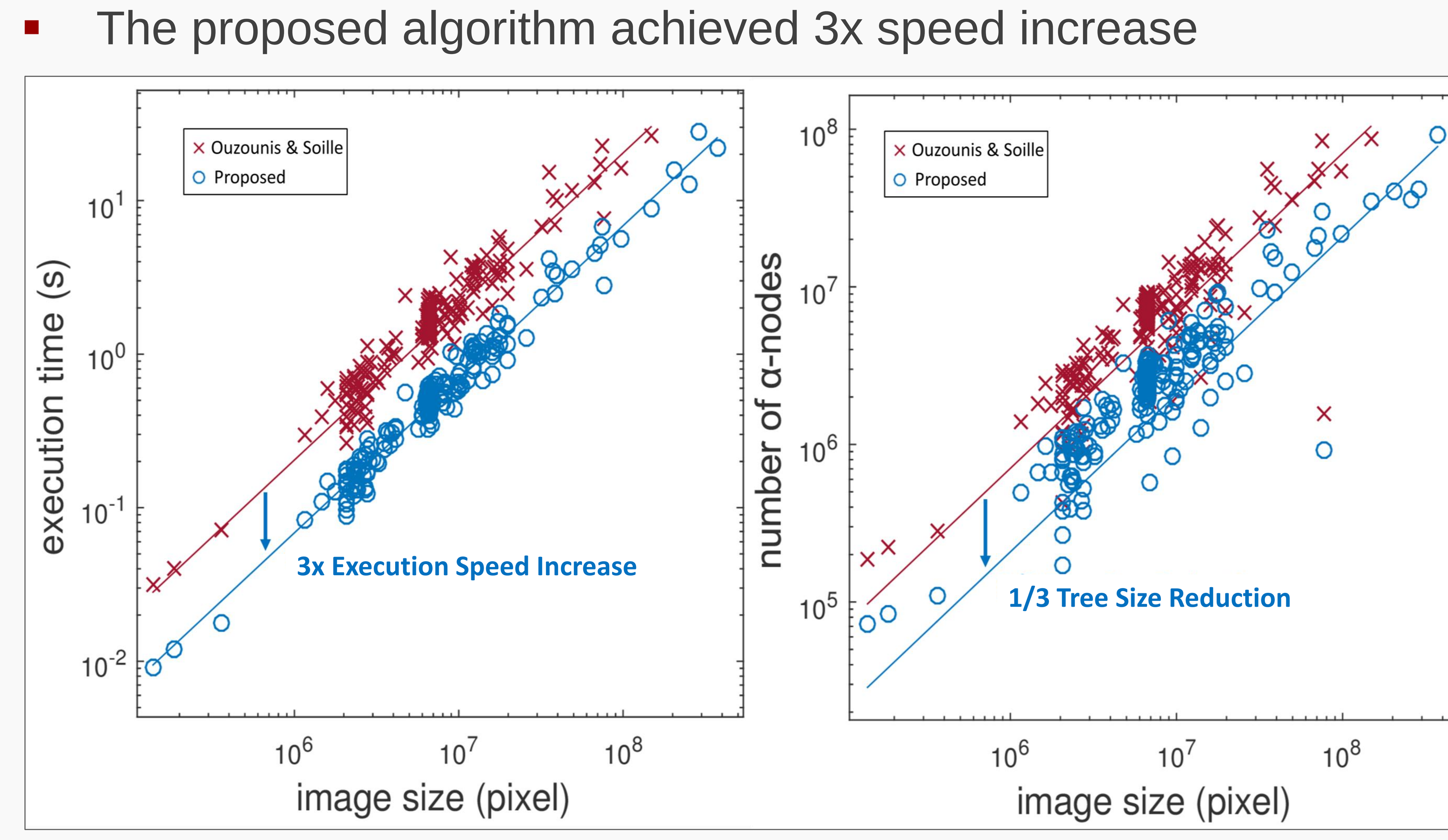
The α–TSE Modeling

- The α–TSE model was optimized to maximize memory efficiency
- Confidence Interval (95%) of TSE model error was only 5.8% of the maximum tree size (*N*)



Execution Speed Improvement by Flooding

- The proposed algorithm was compared to Ouzounis–Soille's [3]
- The proposed algorithm achieved 3x speed increase



Conclusion

- The proposed α–tree algorithm achieved 3x execution speed increase
- The proposed α–tree algorithm reduced the memory use by 41%
- We modeled the α–tree size using an exponential decay function
- We will apply the α–TSE to pilot max–tree of astronomical images in [2]

[1] Teeninga, Paul, et al.: International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing (ISMM), pp. 157-168 (2015).
[2] Moschini, U., Meijster A., and Wilkinson M.H.F.: IEEE transactions on pattern analysis and machine intelligence 40(3) 513–526 (2018)
[3] Ouzounis, G.K., and Soille P.: JRC Technical Reports, Joint Research Centre, European Commission (2012)
[4] Wilkinson M.H.F.: 2011 18th IEEE International Conference on Image Processing (ICIP), pp. 1021–1024 (2011)