

AUTOMATED COLOR AND LOCATION COMPARISON AS A TIMESTAMP RECOGNITION METHOD  
IN IMAGES

by

JEREMY WEGRZYN

B.M., University of Massachusetts Lowell, 2011

A thesis submitted to the  
Faculty of the Graduate School of the  
University of Colorado in partial fulfillment  
of the requirements for the degree of  
Master of Science  
Recording Arts Program

2017

This thesis for the Master of Science degree by

Jeremy Wegrzyn

has been approved for the

Recording Arts Program

by

Catalin Grigoras, Chair

Jeff Smith

Scott Burgess

Date: May 13, 2017

Wegrzyn, Jeremy (M.S, Recording Arts Program)

System for Automated Detection and Recognition of Timestamps in Still Images

Thesis directed by Assistant Professor Catalin Grigoras

### **ABSTRACT**

A method to recognize a source camera or family of cameras from a photograph's timestamp would be a useful tool for image authentication. The proposed method uses the color and location of the timestamp in a photo of dubious origin and compares those characteristics to the known values of various camera makes and models. This method produces accurate results for the test images, correctly recognizing photos from families of cameras with known characteristics and excluding incorrect families. There are challenges to the recognition when the test image has been compressed heavily or reduced in resolution, as happens when uploaded to social media platforms. By looking at additional characteristics to compare with, such as spacing between characters, and a larger database of comparison data, this method would become an even more useful step in image authentication.

The form and content of this abstract are approved. I recommend its publication.

Approved: Catalin Grigoras

## **ACKNOWLEDGEMENTS**

I would like to thank Dr. Catalin Grigoras for the original algorithm used as the basis for the program developed for this thesis.

**TABLE OF CONTENTS**

CHAPTER		
I.	INTRODUCTION.....	1
	Visual Timestamps as a Means of Camera Recognition .....	1
	Literature Review .....	2
II.	METHODS.....	4
	Timestamp Color and Location Detection .....	4
	Testing Procedure.....	4
III.	RESULTS .....	7
	Known Camera Sources.....	7
	JPEG Recompression, Resolution Adjustment, Pixel Density, and Social Media.....	7
	Unknown Camera Source .....	10
IV.	DISCUSSION.....	14
	REFERENCES .....	18

**LIST OF FIGURES**

1.1 Timestamp from Casio EX-S12 camera .....	2
1.2 Timestamp from BenQ DX E820 camera .....	2
3.1 Recompression Image A - Corn.....	8
3.2 Recompression Image B - Lines.....	8
3.3 Output of Control Image for Sony DSC T33 Camera.....	11
3.4 Output of Image "Evidence.jpg" .....	12
3.5 Image from an Unknown Source .....	12
3.6 Timestamp from Samsung ST60 .....	13
3.7 Timestamp from DC E820 .....	13

**LIST OF TABLES**

3.1 Recognition Results of Unaltered Images from Known Sources.....	9
3.2 Recognition Results of Recompressed and Resized Images .....	10

## CHAPTER I

### INTRODUCTION

#### **Visual Timestamps as a Means of Camera Recognition**

Timestamps printed onto photographs have two qualities that make them potentially useful as a tool in the recognition of a camera make and model. The first quality is the immutability of the timestamp when printed. When an image is printed to a file by a camera or other software with a visual timestamp, that timestamp is embedded destructively into the file. The information "underneath" the stamp is gone. As with any other part of the image, the timestamp cannot be changed except by editing tools and techniques. The second quality is that timestamps are added by specific, repeatable processes. The method by which the timestamp is printed is controlled by the settings of the camera or program that added it. That means that the character placement, the font color, the spacing, the date/time formats, and other characteristics are all generated predictably according to the settings of the camera. Furthermore, the specific characteristics of these settings can vary between camera models and makes, as shown in Figures 1.1 and 1.2. If these characteristics vary between camera models and not at all between identical models with identical settings, it is possible to use these differences as a means of camera recognition for image authentication purposes.

The general process would be to measure or otherwise identify the timestamp characteristics of an image with an unknown source, compare these characteristics to a database of known cameras, and reduce the list of possible sources by excluding those models whose timestamps do not match. This study focused on developing an automated method to analyze two characteristics, timestamp color and placement, of an image as a means of recognition. The method works well in its current state to accomplish this task, but has potential for improvement. It is able to distinguish between different camera makes effectively. For example, it can reliably distinguish between a Sony DSC-T33 camera and a Nikon E8700. It

has difficulty, however, distinguishing very similar timestamps from each other. It was unable to differentiate the timestamp from a Nikon Coolpix L820 from that of a Nikon Coolpix E7600. This method could be improved through several means. With a larger database of timestamp characteristics to compare with, subtle differences in similar timestamps could be distinguished more easily and determine larger groupings of timestamp "families". Additional testing parameters could be added on, such as the spacing between characters, to further reduce the list of potential sources.



**Figure 1.1** Timestamp from Casio EX-S12 camera.



**Figure 1.2** Timestamp from BenQ DX E820 camera.

### **Literature Review**

Searches in online journals for "timestamp recognition" and its variations yielded few results dealing with visible timestamps rather than metadata stamps. The largest concentration of relevant topics was found in the IEEE database. Several of these featured methods for the automatic detection of timestamps. None of the articles found address the concept of matching a timestamp to a source camera.

The proposed method for detection relies upon hard-coded values for recognition, but the detection techniques discussed in the papers by Shahab et al. and Garcia and Aspostolidis could prove useful in future developments. By including character information in addition to the color and location information already used, the recognition could function better. The shape of the characters found in the timestamp could be compared to a known library. This would have the added benefit of giving a more accurate position of the timestamp location comparison. This detection technique could also be applied to determine the black space between characters. As with the other characteristics, the black space in a timestamp is predetermined by its font settings. If the characters could be correctly identified and the black spaces between them measured, that information could be added as another layer of the recognition process.

One unusual resource found is a project from the University of North Carolina Chapel Hill by Stephen Guy. It is a program developed to automatically detect timestamps and "remove" them by means of a content-aware fill. Ignoring the removal aspect, the detection system is partially based on the same elements used by the method proposed in this paper. In addition to color and location, the program by Guy utilizes color saturation and gradient magnitude as part of the detection algorithm. It looks for certain criteria in those four elements, determines a likely location of the timestamp, and then evaluates the mean and standard deviation of the colors in the selected area to identify which pixels belong to the timestamp. This all occurred without additional input from a user or reassigning values for the detection parameters for each model. The detection worked very well, though it did suffer from errors in certain cases. These same methods could be integrated into the proposed method as a way to further automate the detection of a timestamp, even if defined characteristics are still required to compare it to.

## CHAPTER II

### METHODS

#### Timestamp Color and Location Detection

The proposed method for recognition, initially developed by Dr. Catalin Grigoras, relies on color and location to recognize an image as belonging to a specific family of cameras and consists of the following steps:

1. Compare the RGB values of the pixels in an image to a specified range of acceptable RGB values of a known timestamp.
2. Record the x- and y-coordinates of any pixel with RGB values within this specified range.
3. If no color match has been found, the comparison for that camera model terminates with a result of no recognition.
4. If color match has been found, define a range containing all detected pixels using the recorded Cartesian coordinates during step 2.
5. Compute the ratio of the recognized timestamp's vertical placement to the maximum y-resolution of the image.
6. Compare this to the ratio derived from the same process on an image with a known timestamp.
7. If these ratios are equal, the result is a potential recognition for the camera model. Otherwise, the result is no recognition.

#### Testing Procedure

Because of the use of predefined characteristic information, this process must be repeated for each camera model tested for. For this study, the twenty-two images were tested against the characteristics of five different camera models: BenQ DC-E820, Casio EX S12, Nikon Coolpix E8700, Samsung ST60, and Sony DSC-T33. Twelve of these images originated from these five camera models. The remaining ten were from other camera models that did not have

recognition functions defined and were compared to those same five camera models previously named. Additionally, one image from a Nikon Coolpix E8700 was altered several times to disrupt the recognition process. In two versions of the image, lines or shapes were drawn on that matched the color of a different, recognizable timestamp. In three others, the timestamp of another recognized camera model was copied and pasted onto the image in various positions: one next to the actual timestamp, one overlapping the actual timestamp, and one placed in its expected position, completely covering the actual timestamp.

Three images from a Nikon Coolpix E8700 were uploaded to Facebook and downloaded again to see what effect compression and resolution changes commonly seen on social media platforms have on the recognition process.

To further test compression and resolution effects, these same three images were resaved with different jpeg compression levels, resolutions, or both using the image editing program GIMP, version 2.8.

The compression recognition tests were run on two Nikon Coolpix E8700 images starting with a quality setting of 90 on GIMP's jpeg export screen. This setting was decreased by 10 to determine if, at any point, the timestamp was no longer consistently recognized. When this cross-over point was found, the setting was then changed in smaller increments to determine if there was an exact number at which the recognition began to fail.

For the resolution tests, a different image from a Nikon Coolpix E8700 was resized several times, preserving the aspect ratio of 4:3 present in the original image. This was again performed in GIMP 2.8, with a quality setting of 100 for each image. It was resized to several standard resolutions, such as 800x600 and 640x480, to again see if there was a crossover point wherein the recognition became inconsistent. As with the compression tests, when this crossover was found, the resolution changes were made in much smaller intervals.

As an additional difference to explore, the same image's pixel density was changed, ranging from its original three-hundred dots-per-inch down to ten, reducing it by approximately fifty percent each time.

The last test performed for the social media-style changes, an image was created with a resolution of 1224x918, a dpi of 96, and a quality of 72. These were selected as the closest determinable elements of an image downloaded from Facebook. The resolution and dpi of the image were made to match the original version that had been downloaded from Facebook. The quality of 72 was a best guess at the actual compression value, which could not be exactly determined. This was chosen by trial and error with respect to file size. The image at resolution 1224x918 and 96 dpi was exported at various quality levels to determine which was the closest to the original image's file size of 283 kilobytes. The created image is 281 KB, so it is not a perfect recreation. Ideally, this would have been created using the exact compression level.

In addition to the known-source and altered images, eighteen images from unknown camera sources were tested against the same five cameras used during the known-source testing. The camera sources of these images could not be determined from their exif data, but, based on the visual appearance of their timestamps, these images appear to be from no fewer than nine different camera types. These tests are intended to act as a simulation of the various conditions this system may encounter in actual use – images with different resolutions, some having recompression issues, analog timestamps rather than digital.

## CHAPTER III

### RESULTS

#### **Known Camera Sources**

The twelve images from recognizable camera models were all reported as coming from their respective, known sources and did not report any additional, false models.

Nine of the eleven images from non-recognizable models reported no models recognized. Two images, originating from a Nikon Coolpix L820, reported as containing the timestamp of a Nikon Coolpix E8700. No other false positive results were returned.

The images altered specifically to disrupt recognition had similarly positive results. The images with lines and shapes of a known timestamp color reportedly recognized a Nikon Coolpix E8700, their original source, and no other types recognized. The images wherein a different, recognizable camera's timestamp was pasted next to or partially obscuring the original reported only the source camera model, not that of the pasted timestamp. When the fake timestamp was pasted in its expected position, completely obscuring the original, no recognition was reported for any camera model.

#### **JPEG Recompression, Resolution Adjustment, Pixel Density, and Facebook**

The JPEG recompression for Image A, shown in Figure 3.1, had positive results above quality level twenty-seven for image. For all qualities above that, with the exception of quality level forty, the timestamp from the correct camera was recognized and no incorrect were. Quality level forty had no camera recognized. At all qualities twenty-seven and below, no timestamps were recognized. Image B, shown in Figure 3.2, was recognized successfully at all quality levels tested. Unfortunately, it is not clear what "quality level" means as GIMP uses it, so it is difficult to determine how this may compare to compression from other sources, such as social media platforms.



**Figure 3.1.** Recompression Image A - Corn



**Figure 3.2.** Recompression Image B - Lines

The resolution adjustment tests had varied results. At all resolutions 900x675 and above, the correct camera type, Nikon Coolpix E8700, was correctly identified. Between resolutions of 732x549 and 800x600, the system recognized both the Nikon and a Casio EX S12. There was one exception within this range, at resolution 735x551, where only the Nikon was recognized. In images with resolutions below 720x540, the recognition were even less consistent. In some cases, it would recognize the correct model. In others, it would recognize only an incorrect model or none at all.

The correct timestamp was recognized in all images with altered pixel densities. None of the images uploaded to Facebook had any cameras recognized. The fake Facebook image, an image altered to match a Facebook upload as closely as possible, was successfully recognized, despite having the same original image, the same resolution, the same file type, and roughly the same file size.

**Table 3.1** Recognition Results of Unaltered Images from Known Sources

Camera Model	Number of Images Tested	Recognized Camera	Result
BenQ DC-E820	2	BenQ DC-E820	Correct Identification
Casio EX S12	1	Casio EX S12	Correct Identification
Kodak Easyshare V1003 Zoom	1	None	As expected
Nikon Coolpix E8700	7	Nikon Coolpix E8700	Correct Identification
Nikon Coolpix L820	2	Nikon Coolpix E8700	Incorrect Identification
Nikon Coolpix E7600	1	None	As expected
Nikon D70	1	None	As expected
Panasonic Lumix GH2	2	None	As expected
Polaroid PDC 3070	1	None	As expected
Samsung ST60	1	Samsung ST60	Correct Identification
Sony Cyber-shot DSC P10	1	None	As expected
Sony DSC-T33	1	Sony DSC-T33	Correct Identification
Spy Tec Inventio-HD 720P	1	None	As expected
Eastman Kodak DX 6340	1	None	As expected

**Table 3.2** Recognition Results of Resized Images and Recompression of Image A

	<b>Camera Recognized</b>	<b>Result</b>
<b>Resolution Altered- Resolution</b>		
560x420	Casio EX S12	Incorrect Recognition
640x480	None	No recognition
720x540	None	No recognition
725x544	Nikon Coolpix E8700	Correct Recognition
730x548	Casio EX S12	Incorrect Recognition
732x549	Nikon Coolpix E8700, Casio EX S12	Two recognized, one correct
735x551	Nikon Coolpix E8700	Correct Recognition
740x555	Nikon Coolpix E8700, Casio EX S12	Two recognized, one correct
760x570	Nikon Coolpix E8700, Casio EX S12	Two recognized, one correct
800x600	Nikon Coolpix E8700, Casio EX S12	Two recognized, one correct
900x675	Nikon Coolpix E8700	Correct Recognition
960x720	Nikon Coolpix E8700	Correct Recognition
1224x918	Nikon Coolpix E8700	Correct Recognition
<b>Recompression- Quality Level of Image A</b>		
0	None	No recognition
10	None	No recognition
20	None	No recognition
25	None	No recognition
27	None	No recognition
28	Nikon Coolpix E8700	Correct Recognition
29	Nikon Coolpix E8700	Correct Recognition
30	Nikon Coolpix E8700	Correct Recognition
40	None	No recognition
50	Nikon Coolpix E8700	Correct Recognition
60	Nikon Coolpix E8700	Correct Recognition
70	Nikon Coolpix E8700	Correct Recognition
80	Nikon Coolpix E8700	Correct Recognition
90	Nikon Coolpix E8700	Correct Recognition

### Unknown Camera Sources

Of the twenty images from unknown sources, fourteen images were not recognized as having any of the known timestamps. This was the expected result for eight of these images as they had timestamps that, visually, appeared inconsistent with the five timestamp families known to the system. Each of the remaining six appeared consistent with one the five families. These six images were also on the lower end of the acceptable resolution levels, so image quality may have skewed the recognition.

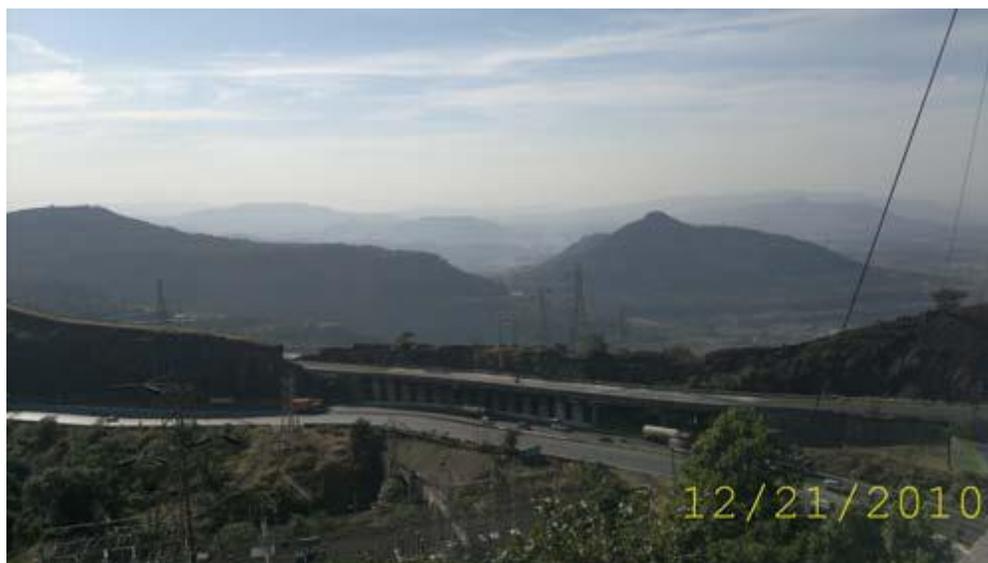
Of the five images that recognized at least one of the five timestamps, three appeared to be false positives. The detection algorithm highlighted unexpected areas of the photographs as containing a timestamp and was able to pass both the color and location detections. These three were of lower resolutions and may have been affected by this. One image, shown in Figure 3.3, recognized as belonging to a Sony DSC T33, which appeared visually consistent with the DSC T33 timestamp, shown in Figure 3.4. The final image, shown in Figure 3.5 had two cameras recognized, which had not occurred for any other image tested, a Samsung ST60 and a DC E820. The ST60 timestamp, in Figure 3.6, does not appear visually consistent nor does the region highlighted as containing this timestamp. A DC E820, showing in Figure 3.7, could be a potential source depending on the settings of this camera or could exist in the family of cameras that this image originated from.



**Figure 3.3.** Output of Image "Evidence.jpg"



**Figure 3.4.** Output of Control Image for Sony DSC T33 Camera



**Figure 3.5.** Image from an Unknown Source



**Figure 3.6.** Timestamp from Samsung ST60



**Figure 3.7.** Timestamp from DC E820

## CHAPTER IV

### DISCUSSION

While the success rate for recognitions is high for the known camera types, the results of the manipulation tests and the unknown source tests highlight certain challenges with the method as a whole or, at least, its current implementation that will need to be overcome to create a consistently accurate technique for timestamp recognition. The inconsistencies of the resolution and compression tests show that there are certain criteria that, at this point, images must meet in order to return accurate results.

The first criterion is that images need to have a sufficiently high resolution. Based on the results collected, the minimum required resolution seems to be around 960x720. With resolutions smaller than this, inconsistencies with recognition occur more frequently. In a high resolution image, it is easier to find an "average" pixel color for the timestamp and there are more likely to be pixels within a small range of that color. If an image has a lower resolution, the acceptable range of colors for the timestamp has to be wider due to the loss of information. This has the undesirable effect of decreasing the accuracy of the color detection. A wider variety of pixel values will trigger the detection, increasing the likelihood of false positives generating and distorting what timestamps may be detected and where. This effect can be seen in the control image for the Sony DSC T33, as seen in Figure 3.4, though it is created by a different process than resolution adjustment. That image was an average of images from a DSC T33 with partial transparency. Because of the overlap of some characters, but not all, the search algorithm was tuned to accept a wider range of color values. While not extending beyond the timestamp in this case, this imprecision causes inaccuracy in other images from Sony DSC T33 cameras, as seen in Figure 3.3. It is important to note that this image is only suspected to be from a Sony DSC T33 as the actual camera source is unknown.

The second criterion is that the image must either be lossless or not have too high of a compression ratio. This is, however, an inexact criterion. The affect of compression on recognition appears to be heavily dependent on the content of an image. In one case, Figure 3.1, the image quality affected recognition substantially, with no successful results at a quality below 28. Meanwhile, in the other case, Figure 3.2, the Nikon Coolpix timestamp was successfully recognized at quality zero. Compression has an effect on recognition, but this can be offset. There are two likely elements that offset compression: resolution and image content. The image in Figure 3.2 was resized to resolution 2592x1944, smaller than the original, to match the resolution of Figure 3.1's image. The recognition of this resized image of lines was unsuccessful at low quality levels, but still required heavier compression to reach the failure point than the image of corn. This could be due to content of the images. The corn image is much more dynamic in terms of shapes, color range, and contrast. The lines image is composed of primarily two colors outside of the timestamp and consists of a largely repeating pattern. The substantial color and contrast difference may be the reason one image could be successfully recognized at low quality levels while the other could not. Rather than the high compression blending the timestamp into the background, as with the corn, the information remains highly differentiated.

Even with the criteria met, this process, as with most forensic tools, is not effective as a singular approach in image authentication and is intended for use as part of a larger toolbox or framework. There are several challenges that keep timestamp recognition from being able to determine exactly what the source camera of an image is. The principal difficulty is that many camera models may share the same or very similar templates for timestamps. This is suspected to be the cause of the misrecognition of the Nikon L820 and with the unknown source image that looked like it could have been from a DC E820. If multiple cameras create very similar, if

not identical timestamps, the ability to determine an exact source becomes effectively impossible. However, there are potential solutions to this problem.

The exact camera source does not necessarily need to be determined. As stated above, this process is intended to be used as a step for image authentication. The authentication process relies on the question “is this quality consistent?” This method can give an answer that question about a image’s timestamp, and will do so even better with further development. It does not need to be able to pick the exact model from a list and no others. It is enough to include the model or the family of models as possibilities in order to support a conclusion and rely on the other steps of the authentication process to further support it.

Development of a wider database of timestamp information is necessary for this to function. The more timestamp information that is available, the more complete the list of potential sources becomes. As was seen with the known and unknown camera source results, having a database of five camera models was insufficient. The majority of results were no recognition, which has limited value in the authentication process. By collecting a large body of data with greater variety of camera models, the system can determine the list of potential sources far more accurately. This gives the entire method more weight in the authentication process and, therefore, acts as greater support to the derived conclusions.

It would also be helpful to compile a database of different settings for otherwise known timestamps. Cameras often have different settings available for timestamps, such as changing format of the date and time. By collecting information about how these settings affect timestamp characteristics, the recognition process can be made even more accurate. It can further narrow the list of potential sources based on what cameras have the detected settings.

There are also improvements that can be made to the process to make it more accurate. There are several different timestamp characteristics that could be utilized beyond color and location. The text recognition described in the literature review could be a useful addition. With

a sufficiently large database of timestamp font information, each individual character could contribute to the recognition process by the comparison of character size, shape and location, for characters with lower variation. This also opens up the possibility of using character spacing as part of the recognition process by analyzing the distance between characters and comparing it to either an average or to known distances. This could potentially be implemented without specific character recognition by averaging the distance between two characters as long as at least one is fixed. As seen in the averaged image in Figure 3.4, certain characters are in the same place in each image. By determining the average distance between this fixed character and the expected characters around it, one could potentially narrow the list of recognized sources even further. The methods developed by Stephen Guy, specifically gradient and saturation detection, could further refine the detection and comparison processes already developed.

## REFERENCES

Schatz, Bradley, George Mohay, and Andrew Clark. "A Correlation Method for Establishing Provenance of Timestamps in Digital Evidence." *Digital Investigation* 3, Supplement (September 2006): 98–107.

Guy, Stephen. "Computational Photography Final Report - Automatic Timestamp Detection and Removal from Digital Photos." University of North Carolina Chapel Hill, December 5, 2008. <https://wwwx.cs.unc.edu/~sguy/CompPhoto/>.

Shahab, Asif, Faisal Shafait, and Andreas Dengel. "Bayesian Approach to Photo Time-Stamp Recognition." IEEE, 2011. doi:10.1109/ICDAR.2011.210.

Chen, Xiangrong, and Hong-Jiang Zhang. "Photo Time-Stamp Detection and Recognition." IEEE, 2003. doi:10.1109/ICDAR.2003.1227681.

Garcia, C., and X Apostolidis. "Text Detection and Segmentation in Complex Color Images." IEEE, 2000. doi:10.1109/ICASSP.2000.859306.