OptiColor Software Tool

Purpose of the Tool

Visual Search research has seen numerous research works on the concept of increasing worksite conspicuity. An intuitive approach to achieve it is by maximizing the contrast between a HVSA and the ambient worksite colors. This approach when used appropriate has the possibility to improve accuracy and speed performance. This is primarily due to the wearer of HSVA being more conspicuous and identified more quickly by others.

Our algorithm recommends the most appropriate HVSA color based on the worksite using rigid mathematical formulas. In order to validate our recommendation in real scenarios, we have implemented a software tool that measure response times of a point and click visual search of our dataset. The dataset consists of worksites images that entail people with randomly chosen HVSA color and the same images edited with our recommend HVSA color. Our goal is to see the difference in response time and accuracy by volunteers in the two setups.

The OptiColor Image Dataset

To perform our experiment we collected 15 images of three different worksites pertaining various number of personnel. These images where then doubled by editing the HVSA apparel of the personnel to match the optimal HVSA color determined by our approach. This resulted in the final dataset with 90 images. We dub the initial pictures as the Original set and the edited ones as the Adapted set.

Additionally, we annotated the images with bounding boxes depicting the area occupied by a person. This information is used by the software tool for depicting the clicked areas and for calculating the relevant metrics. The total number of personnel on the 45 images is 167, thus an average of 3.7 people per image.

The OptiColor Software Tool

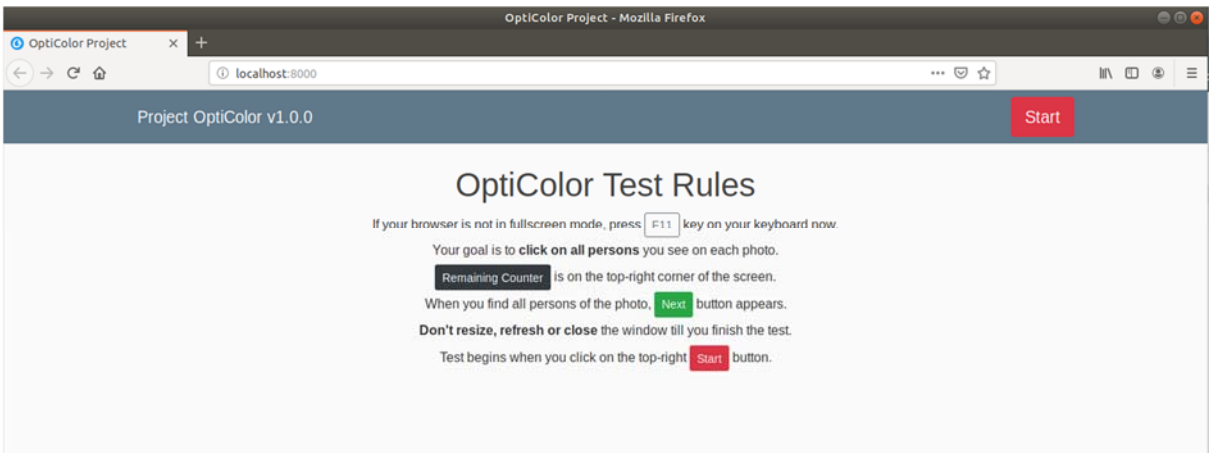
A web application has been developed using the Django web framework to:

- randomly present to the volunteers worksite images,
- record their mouse clicks on the images
- store the relevant information into a database.

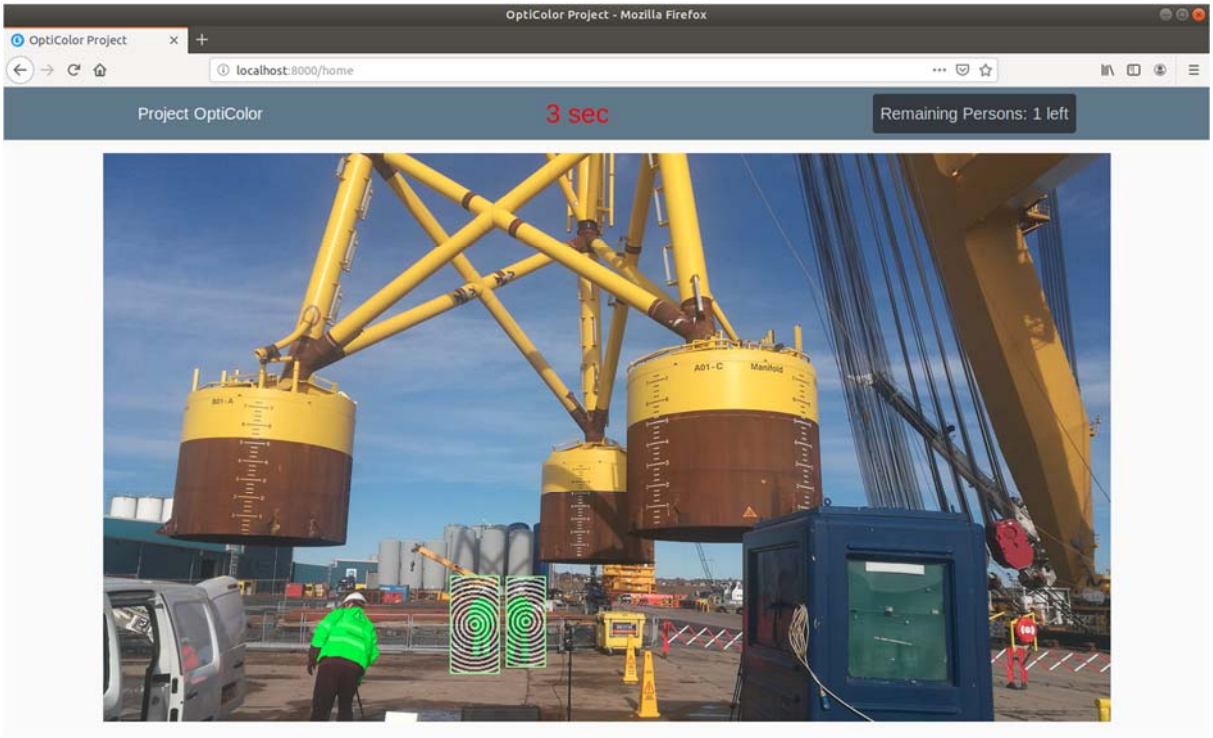
Software Tutorial

The software runs locally on the computer with the only prerequisite to have installed Python 3.7.3. We include a script for running in Windows and Linux. For Windows a double click on the shell script will initialize a local server and open your default browser to the initial web page of the tool (see image 1). For Linux Ubuntu you need to open a terminal window on the folder location of the tool and type $ source run_ubuntu.sh this will in effect execute the same functionality as the windows shell script.

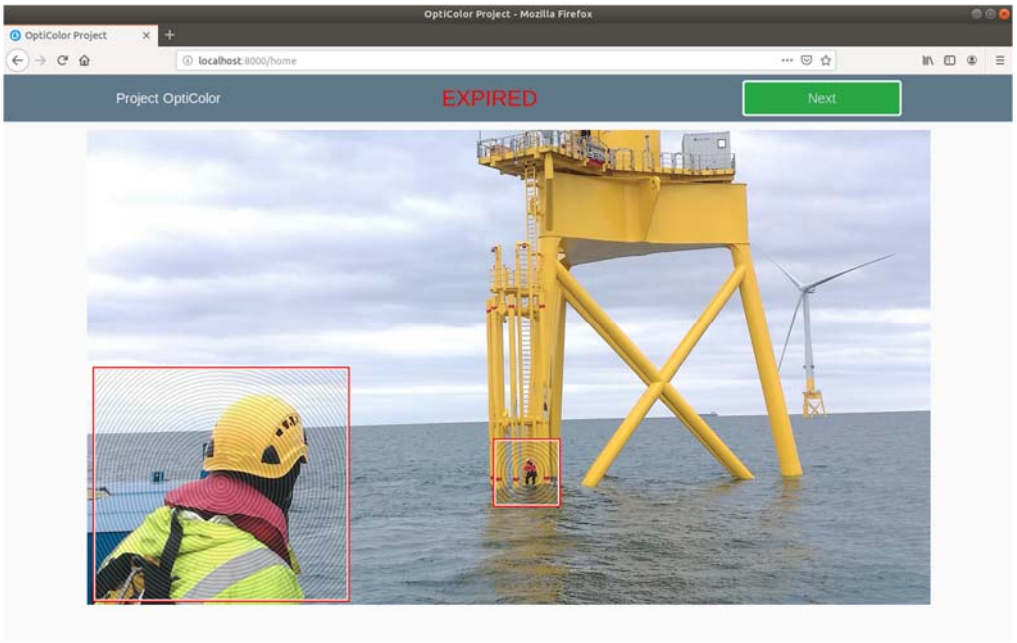
Note: The software works flawlessly without any supplementary libraries because all of the required additions are included in the built in virtual environment.



After we initialize the web application we can perform a test run of the application. As seen at Image 1 some details of the test are presented to the user. When we have familiarized with the rules of the test, we can start the test by pressing the ‘start’ button. The software will then start presenting to us randomly chosen pictures from our dataset (see Image 2).

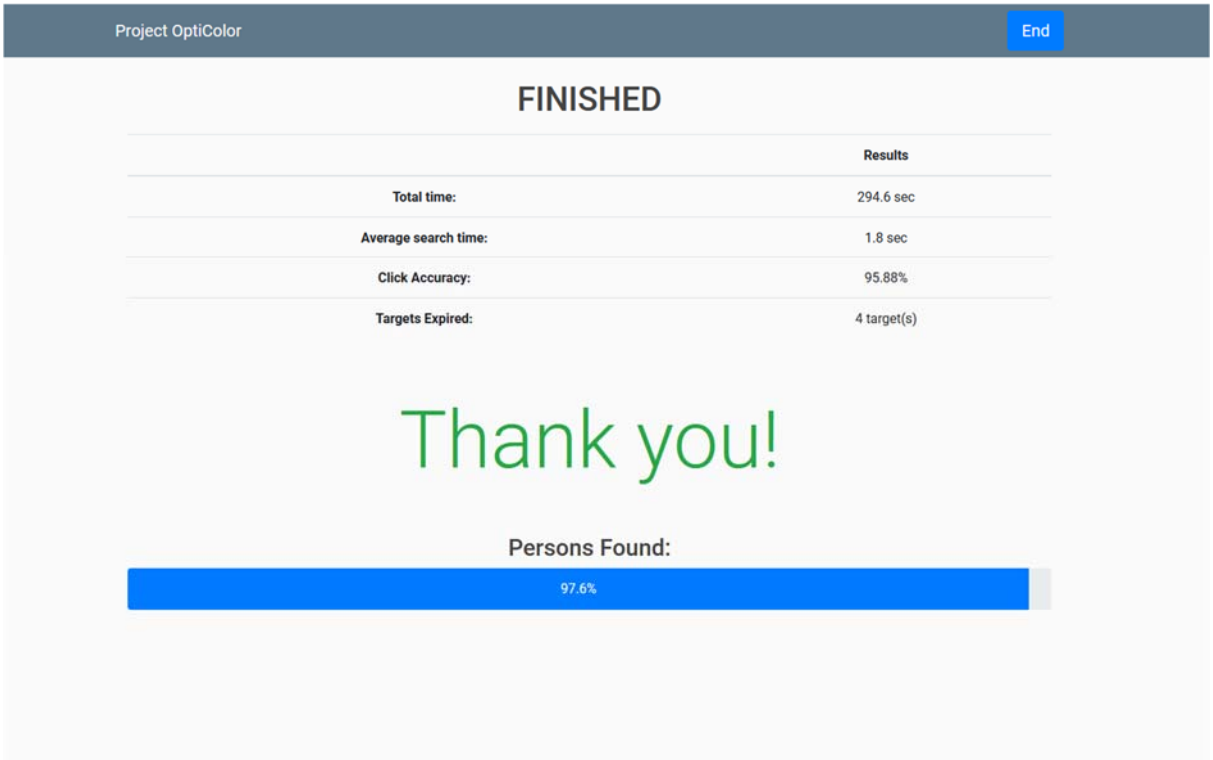


We can perform visual search and click using the computer mouse the personnel on the worksites. When we click on a person the area occupied by the person is highlighted and excluded from the clickable area in order to avoid duplicate clicks. When we click a wrong area the image shakes slightly horizontally. Additionally, in the top right corner we can get some information regarding the remaining persons and on the top middle area we can see the remaining time for finding the personnel. The tool allows of a time frame of 4 seconds per person for the visual search. In the scenario where the timer runs out, we are presented with the people that we didn't manage to find (see Image 3).



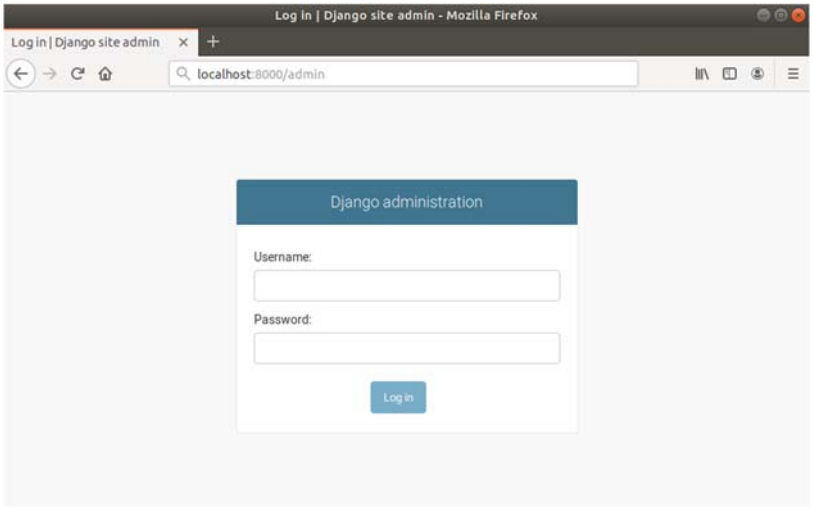
When we have found all the personnel or the timer has run out, a 'Next' button is displayed

in the top right corner, to continue with the next image. The presented pictures are shown in a random order with the only constraint that when an image of the Original or the Adapted set is presented then the corresponding image on the alternative set is excluded. Thus the tool presents 45 utterly different images to the user and ensures that no prior knowledge of the person's position is ever exploited by the volunteer. When the experiment is completed, the tool displays some general information such as time spent on the experiment, time between clicks (average), user's accuracy and targets missed (see Image 4).

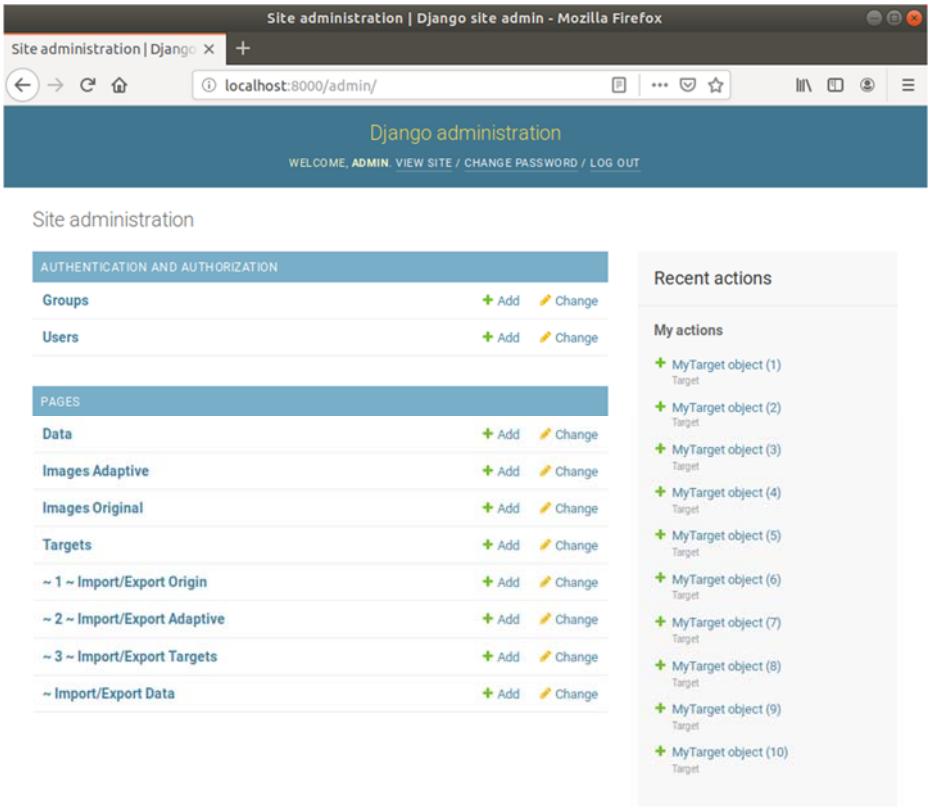


Web Page Administration

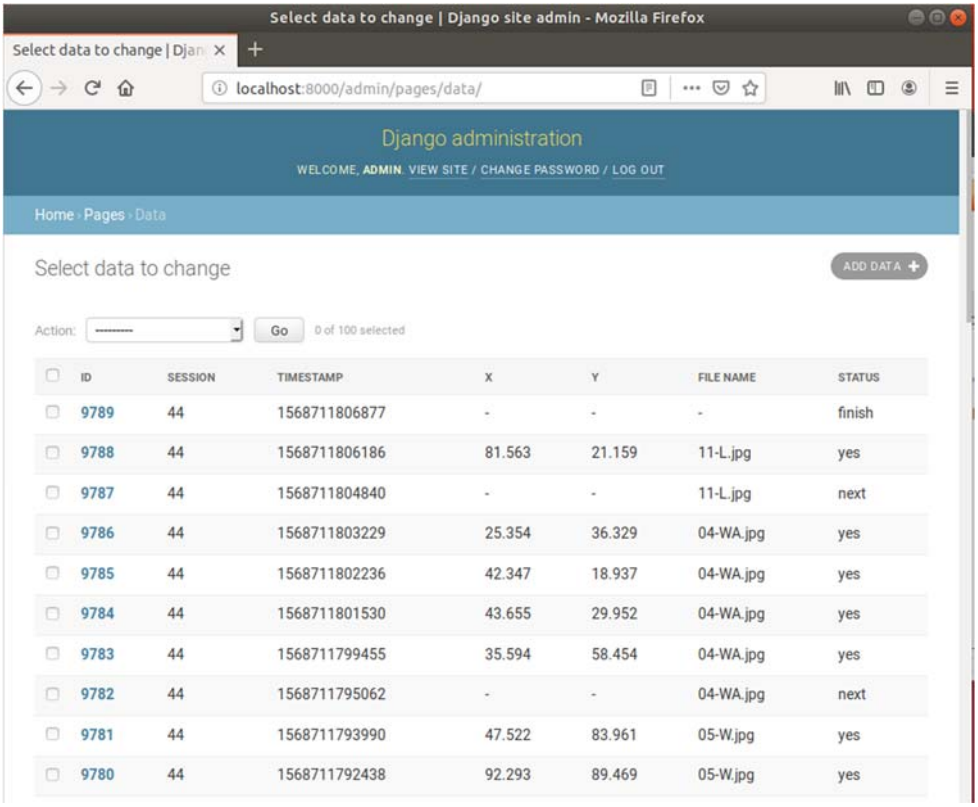
In the background, the tool primarily collects information of the position and the timestamp of the click, but also any additional information required. These information are stored into a database for the accurate computation of the relevant metrics. To access this database, the tool administrator can go to the OptiColor admin page by going to the 'localhost:8000/admin' url (see Image 5) and sign in. The username is 'admin' and the password is 'opticolor'.



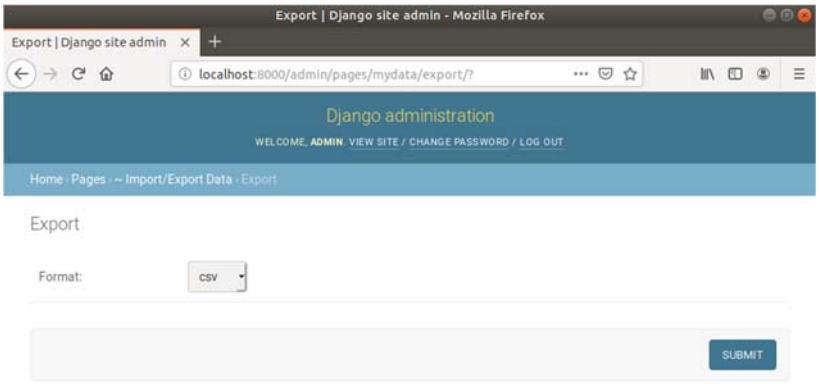
After a successful sign in, we can access various types of information (see Image 6).



At the ‘Data’ page we can see the information collected by the tool during a test (see Image 7).



This information can be exported easily in the ‘Import/Export’ Data page in any desirable format (see Image 8).



Validation Results.

To validate our approach we used the OptiColor software tool on 50 volunteers and computed various performance metrics. To give a general idea of the test, the average time spent to perform the test was 4 minutes and 40 seconds, the average people found was 165.1 (from a total of 167), and there was an average of 11.04 wrong clicks per test. In the following images (1a and 1b), we depict the bounding boxes and associated clicks of all the volunteers in the original and adapted image.

1a - Original image 1b - Adaptive image

Regarding the concept of time, we have observed an average of 6.21 seconds per image where the average time spent on the images of the Original dataset was 7.65 seconds and on the Adaptive dataset 5.01 seconds. This clearly shows a faster response on the visual search on the Adaptive dataset. Specifically, a time reduction percentage of 34.57%.

In order to be sure that the time performance improvement does not have a negative effect at accurately spotting the personnel we calculate accuracy, precision, recall and F1 score on each of the images. To calculate these metrics we need the true positives, true negatives, false positives, false negatives of each image. In more detail, true positives is the number of clicks on the image that were people. False positives is the number of clicks on the image that weren't people. False negatives is the number of people that weren't clicked due to the counter running out. And finally the false positives which is not measurable in our scenario and have a constant value of zero (didn't click on area and it wasn't a person).

Precision is the fraction of relevant instances among the retrieved instances, while Recall is the fraction of relevant instances that have been retrieved over the total amount of relevant instances. Accuracy is used as a statistical measure of how well a binary classification test correctly identifies or excludes a condition. That is, the accuracy is the proportion of true results (both true positives and true negatives) among the total number of cases examined. is a measure of a test's accuracy. F1 scope it considers both the precision p and the recall r of the test to compute the score: p is the number of correct positive results divided by the number of all positive results returned by the classifier, and r is the number of correct positive results divided by the number of all relevant samples.

Accuracy = (TP + TN) / (TP + TN + FP + FN)
Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

F1 = 2 * Precision * Recall / (Precision + Recall)

Detail results on each image are depicted in Tables A1, A2, A3. Images with the suffix “A” on their name are the adapted images.

Table A1: Offshore-Lifting images

	Accuracy		Precision		Recall		F1 score		Search Time		Targets		
	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	All
01-L.jpg	0.9032	0.9362	0.9032	0.9362	1.0	1.0	0.9492	0.967	4.93	4.15	4	4	4
02-L.jpg	0.9841	0.9744	0.9841	0.9744	1.0	1.0	0.992	0.987	4.19	4.61	4	4	4
03-L.jpg	0.974	0.9394	0.9868	0.9394	0.9868	1.0	0.9868	0.9688	7.86	6.46	3.95	4	4
04-L.jpg	0.7089	0.9545	0.8116	0.9545	0.8485	1.0	0.8296	0.9767	11.33	4.57	2.55	3	3
05-L.jpg	1.0	0.9804	1.0	0.9804	1.0	1.0	1.0	0.9901	3.12	2.93	2	2	2
06-L.jpg	0.7977	0.9286	0.8313	0.9369	0.9517	0.9905	0.8875	0.963	12.89	7.60	4.76	4.95	5
07-L.jpg	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	5.52	4.25	4	4	4
08-L.jpg	0.9524	0.9649	0.9524	0.9649	1.0	1.0	0.9756	0.9756	5.16	4.29	5	5	5
09-L.jpg	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	4.04	4.05	3	3	3
10-L.jpg	0.967	0.9912	0.967	0.9912	1.0	1.0	0.9832	0.9956	5.04	4.24	4	4	4
11-L.jpg	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	2.84	2.79	1	1	1
12-L.jpg	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	3.59	3.54	2	2	2
13-L.jpg	0.9818	0.9787	0.9818	0.9787	1.0	1.0	0.9908	0.9892	3.23	2.84	2	2	2
14-L.jpg	0.9547	0.9825	0.9547	0.9825	1.0	1.0	0.9768	0.9912	8.23	7.49	8	8	8
15-L.jpg	1.0	0.9878	1.0	0.9878	1.0	1.0	1.0	0.9939	4.15	3.67	3	3	3

Table A2: Fabrication images

	Accuracy		Precision		Recall		F1 score		Search Time		Targets		
	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	All
01-F.jpg	0.9292	0.9797	0.9292	0.9797	1.0	1.0	0.9633	0.9898	6.74	5.70	5	5	5
02-F.jpg	0.963	0.973	0.963	0.973	1.0	1.0	0.9811	0.9863	3.27	3.27	3	3	3
03-F.jpg	0.837	0.989	0.8828	0.9944	0.9417	0.9944	0.9113	0.9944	20.09	8.24	5.65	5.97	6
04-F.jpg	0.7876	0.9391	0.8091	0.9391	0.9674	1.0	0.8812	0.9686	10.95	5.39	3.87	4	4

05-F.jpg	0.8922	0.9494	0.8976	0.9494	0.9933	1.0	0.943	0.974	8.36	5.83	5.96	6	6
06-F.jpg	0.969	0.9615	0.969	0.9615	1.0	1.0	0.9843	0.9804	5.43	6.15	5	5	5
07-F.jpg	0.9787	1.0	0.9787	1.0	1.0	1.0	0.9892	1.0	3.52	2.75	2	2	2
08-F.jpg	0.7034	1.0	0.7338	1.0	0.9444	1.0	0.8259	1.0	10.65	4.29	3.78	4	4
09-F.jpg	0.7123	0.7231	0.7123	0.7344	1.0	0.9792	0.832	0.8393	5.16	4.75	2	1.96	2
10-F.jpg	0.9882	0.9851	0.9882	0.9851	1.0	1.0	0.9941	0.9925	4.73	4.13	3	3	3
11-F.jpg	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	3.12	2.93	1	1	1
12-F.jpg	0.9483	0.9894	0.9821	0.9894	0.9649	1.0	0.9735	0.9947	5.82	3.60	2.89	3	3
13-F.jpg	0.8806	0.9559	0.8939	0.9559	0.9833	1.0	0.9365	0.9774	11.72	8.78	4.92	5	5
14-F.jpg	0.96	1.0	0.96	1.0	1.0	1.0	0.9796	1.0	3.07	2.49	1	1	1
15-F.jpg	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	2.40	2.53	1	1	1

Table A3: Offshore-Wind images

	Accuracy		**Precision**		**Recall**		**F1 score**		**Search Time**		**Targets**		
	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	Orig.	Adapt.	All
01-W.jpg	0.9857	0.9759	0.9857	0.9759	1.0	1.0	0.9928	0.9878	3.98	3.90	3	3	3
02-Wjpg	0.9375	0.9818	0.9574	0.9818	0.9783	1.0	0.9677	0.9908	4.75	3.41	1.96	2	2
03-W.jpg	0.895	0.963	0.9064	0.963	0.9861	1.0	0.9446	0.9811	22.53	11.48	8.88	9	9
04-W.jpg	0.7708	0.9106	0.8605	0.9412	0.881	0.9655	0.8706	0.9532	13.58	7.88	3.52	3.86	4
05-W.jpg	0.95	0.949	0.95	0.949	1.0	1.0	0.9744	0.9738	6.30	4.90	3	3	3
06-W.jpg	0.8148	0.9529	0.8462	0.9529	0.9565	1.0	0.898	0.9759	7.27	3.97	2.87	3	3
07-W.jpg	0.9286	1.0	0.963	1.0	0.963	1.0	0.963	1.0	5.97	4.14	1.93	2	2
08-W.jpg	0.9767	1.0	0.9767	1.0	1.0	1.0	0.9882	1.0	4.04	3.43	3	3	3
09-W.jpg	0.7831	0.9605	0.8904	0.9865	0.8667	0.9733	0.8784	0.9799	11.08	6.46	2.60	2.92	3
10-W.jpg	0.9474	1.0	0.9474	1.0	1.0	1.0	0.973	1.0	3.76	3.07	2	2	2
11-W.jpg	0.9825	0.9785	0.9825	0.9785	1.0	1.0	0.9912	0.9891	6.15	5.71	7	7	7
12-W.jpg	0.7462	0.9006	0.7816	0.9006	0.9429	1.0	0.8547	0.9477	51.94	17.29	14.14	15	15
13-W.jpg	0.9457	1.0	0.9457	1.0	1.0	1.0	0.9721	1.0	4.91	3.96	3	3	3
14-W.jpg	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	2.51	2.56	1	1	1
15-W.jpg	0.9804	0.9804	0.9804	0.9804	1.0	1.0	0.9901	0.9901	4.43	4.82	2	2	2

In Table B, we depict the average performance metrics on the two sets and their

improvements. To ensure that the results are statistically significant we perform a t-test and report the associated p-value.

Table B: Summary

	Accuracy	Precision	Recall	F1 score	Time
Original Set	0.9204	0.9344	0.9813	0.9561	7.6519
Adaptive Set	0.9693	0.9711	0.9978	0.9838	5.0068
Improvement	+ 5.31 %	+ 3.93 %	+ 1.69 %	+ 2.90 %	- 34.57 %
P-value (<0.05)	0.0018	0.0054	0.0034	0.0021	0.0388

We observe statistically significant performance improvements on all the relevant metrics. This validates our approach and presents the effectiveness of our algorithm on recommending the appropriate HVSA color at a worksite.

Comments - Suggestions

One thing that will help our results is the use of eye-tracker device. Measuring eye positions and eye movement will give us the opportunity to calculate volunteer's focus and make some important graphs about which areas on the photo participant's gaze spent more time.

FAQ

Ο υπολογισμός που δίνει το search time. Είναι ο χρόνος που παίρνει στο χρήστη για να βρει τον κάθε επόμενο εργαζόμενο στην φωτογραφία (δηλ. difference between two consecutive time stamps?) ή είναι κάποιος άλλος χρόνος? Από την πράξη/υπολογισμό που ορίζει το search time, θα γίνει ξεκάθαρη και η έννοια του average search time.

Ο χρόνος που ρωτάς ξεκινάει με το πάτημα στο start/next μέχρι το επόμενο πάτημα σε next/finish. Το search time είναι δηλαδή ο χρόνος που παρουσιάζεται η φωτογραφία στον χρήστη μέχρι να ξαναπατήσει next για την εμφάνιση της επόμενης. Στο database αυτό φαίνεται με το στήλη status του model.Data, όπου κοιτάμε τις διαδοχικές τιμές start/next/finish (όποιο παρουσιάζεται).Το average search time υπολογίζεται από συλλογή όλων των διαφορών των χρόνων της εκάστοτε εικόνας δια το πλήθος που παρουσιάστηκε (πόσες φορές παρουσιάστηκε η εικόνα στα τεστ).