sidia

An Automatic Contrast Validation Approach for Smartphone Themes

Aasim Khurshid
Adriano Gil
Felipe Guimarães
Mikhail R. Gadelha
Everlandio Fernandes

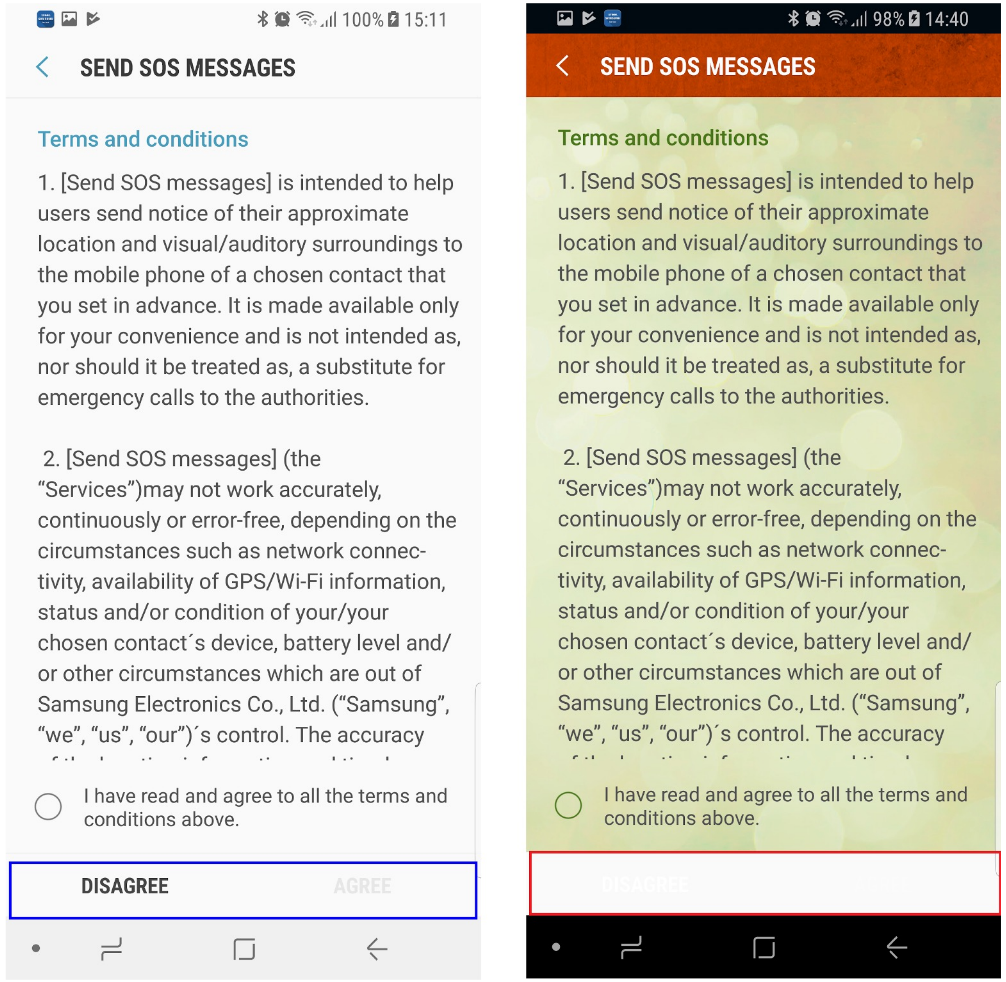
Smartphone Themes

- Customizes more than 100 screens of Android OS UI
- Theme development is an error-prone process that involves configuring more than 200 parameters
- Sometimes a minor change on parameters can generate a visual issue that is hardly detected

Main Challenges

- How to validate the contrast between UI elements?
- The contrast can be described as
  - High Perceivable Contrast (HPC)
  - Low Perceivable Contrast (LPC)

Contrast issues reported by Galaxy Theme Store



Due to a wrong text color, text is not visible on bottom buttons

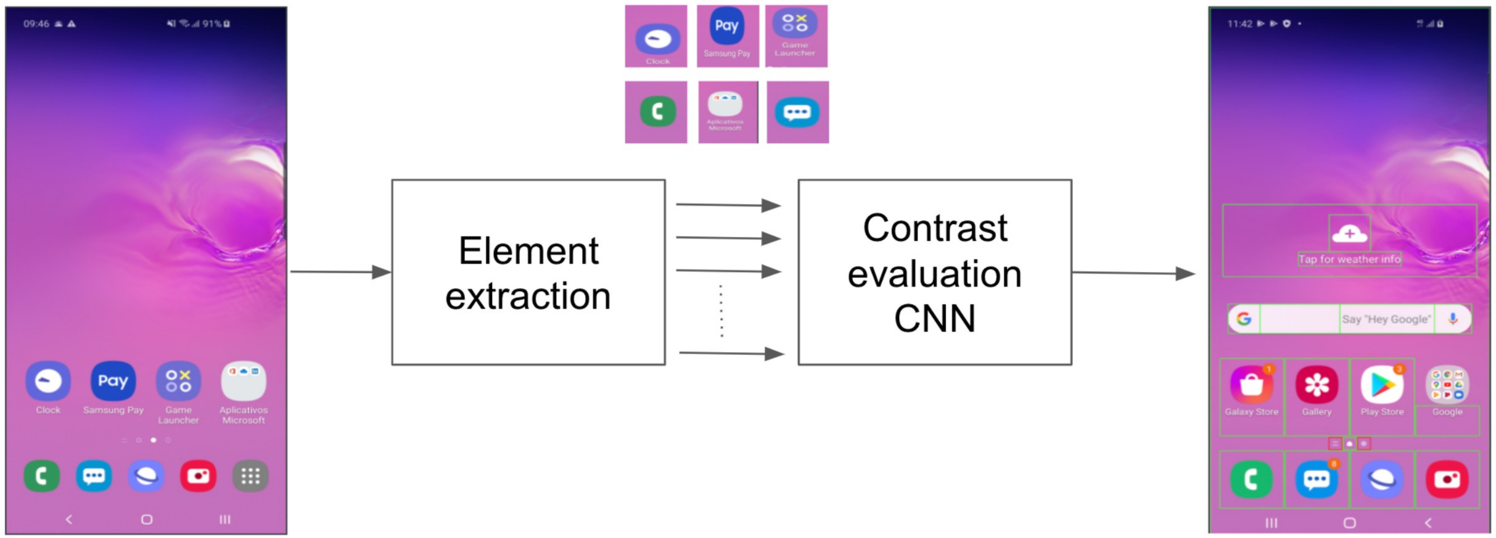
Proposed Approach

- In this work, we propose an automatic themes evaluation approach that validates the contrast of Android smartphone themes among regular (HPC) and non-regular (LPC) at the element level.

Proposed Approach

- To address this challenge, our solution is divided into two phases:
  - **Element extraction:** a mobile application for Android that walks into several screens, takes screenshots and extracts UI information
  - **Contrast Evaluation:** a python-based desktop application that analyzes screenshots and generates a detailed report of any low contrast issues it found.

Proposed Approach



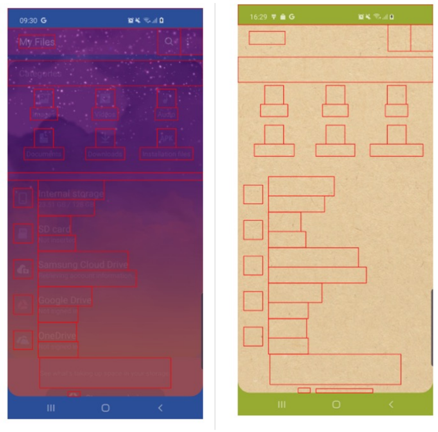
Elements Extraction

- We implemented an Android solution built on top of:
  - UIAutomator: used to simulate inputs
  - Accessibility service: used to extract UI information
- The algorithms in the desktop module **extract the elements** from the screenshot and **send them to our contrast evaluation algorithm**

Contrast Evaluation

- Everlandio et al [6] uses a Faster RCNN with an average precision of 79.91%
- Our approach is based on **Convolutional Neural Network (CNN)**
- **Segmented UI elements** are used as input for our CNN that classifies between two labels:
  - High Perceivable Contrast (HPC)
  - Low Perceivable Contrast (LPC)

Dataset

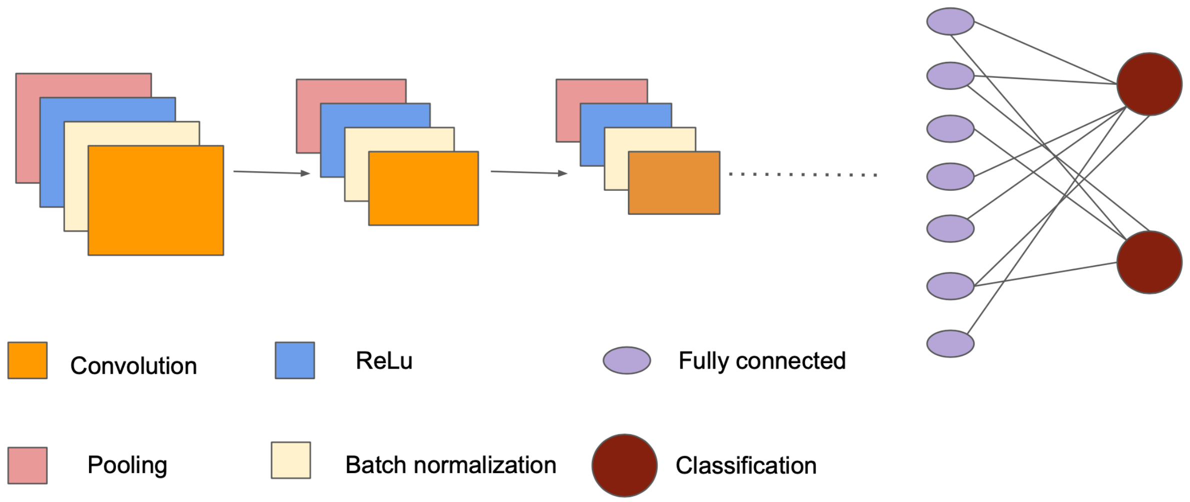


- We created **200 themes** with a wide range of low contrast issues
- For each theme, we captured **122 screenshots**
- **56070 elements** were extracted and manually labeled
- We defined datasets for dark- and light-color themes

Dataset	Dark	Light	Total
Training	29390	26680	56070
Validation	14658	13360	28018
Test	4520	4296	8816

Neural Network Architecture

- Our **Convolutional Neural Network** has only **5 layers**
- In the final part of model architecture there is a fully connected layer followed by a softmax for classification between two labels (LPC and HPC)



Results

- The proposed contrast evaluation network is compared quantitatively with the ResNet50 [17], AlexNet [18] and GoogleNet [19] representative of the state-of-the-art for classification, using the following measures:
    - Attack Presentation Classification Error Rate (APCER)
    - Normal Presentation Classification Error Rate (NPCER)
    - Average Classification Error Rate (ACER)
    - False Positive Rate (FPR)
    - True Positive Rate (TPR)
    - True Negative Rate (TNR)
    - Accuracy
    - F1 score
    - processing time

Results

Method	APCER	NPCER	ACER	FPR	TPR	TNR	Accuracy	F1 Score	Processing time (seconds)
Training and test data extracted from Lightset dataset									
ResNet [17]	0.1821	0.1123	0.1507	0.2150	0.9032	0.7937	0.8445	0.8543	269.01
AlexNet [18]	**0.0048**	0.9956	0.5002	**0.0048**	**0.9956**	0.5002	0.4838	0.3175	184.79
GoogleNet [19]	***0.0241***	**0.0195**	**0.0218**	**0.0241**	***0.9805***	**0.9759**	**0.9781**	**0.9772**	***160.08***
Proposed CNN	0.0517	***0.0239***	***0.0378***	0.0517	0.9761	***0.9483***	***0.9616***	***0.9605***	**56.16**
Training and test data extracted from Darkset dataset									
ResNet [17]	0.2801	***0.0731***	0.1767	0.3721	0.9541	0.6383	0.7895	0.7915	313.24
AlexNet [18]	**0.0625**	0.088	0.0813	**0.0634**	0.9455	0.9032	0.9244	0.9276	183.22
GoogleNet [19]	0.1855	**0.0302**	0.1078	0.1855	**0.9698**	0.8145	0.8887	0.8928	***174.21***
Proposed CNN	0.0817	0.0444	**0.0631**	0.0817	0.9556	**0.9183**	**0.9354**	**0.9314**	**49.08**

Results

- The proposed method has performed consistently well for both the datasets and only has a difference of 0.3 points across datasets.
- Furthermore, the proposed method outperforms all the other methods on the Darkset and is second on the lightset in terms of accuracy and F1 score to GoogleNet [19]

Conclusion

- We presented a solution that traverses Android GUI, automatically takes screenshot and UI information of 122 screens and check for low-contrast issues
- Using a Convolutional Neural Network we achieved a high F1 score and fast processing time
- We concluded that a network architecture with a small number of layers is more able to classify among elements containing or not containing contrast issues.
- In future works we plan to tackle dark and light—colored themes using a single model

References

[6] Fernandes, E., Correia, R., Gil, A., Postal, J., Gadelha, M.R.: Themes validation tool. In: Stephanidis, C. (ed.) HCI International 2019 – Late Breaking Posters (2019)

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, 2012, pp. 1097–1105.

[19] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1–9.

Thank you!

adriano.gil@sidia.com

sidia