Journal of the Brazilian Computer Society, 2024, 30:1, doi: 10.5753/jbcs.2024.3255

A Deep Learning Model for the Assessment of the Visual Aesthetics of Mobile User Interfaces

Adriano Luiz de Souza Lima [**Federal University of Santa Catarina** | ***adriano.lima@ufsc.br***]
Christiane Gresse von Wangenheim [**Federal University of Santa Catarina**| ***c.wangenheim@ufsc.br***]
Osvaldo P. H. R. Martins [**Federal University of Santa Catarina** | ***martins.osvaldo@grad.ufsc.br***]
Aldo von Wangenheim [**Federal University of Santa Catarina** | ***awangenh@inf.ufsc.br***]
Jean C. R. Hauck [**Federal University of Santa Catarina** | ***jean.hauck@ufsc.br***]
Adriano Ferreti Borgatto [**Federal University of Santa Catarina** | ***adriano.borgatto@ufsc.br***]

Department of Informatics and Statistics, Federal University of Santa Catarina, Campus Universitário Trindade, Cx.P. 476, Florianópolis, SC, 88040-370, Brazil.

Received: 20 March 2023 • **Accepted:** 06 October 2023 • **Published:** 04 June 2024

Abstract Visual aesthetics is one of the first aspects that users experience when looking at graphical user interfaces (GUIs), contributing to the perceived usability and credibility of a software system. It can also be an essential success factor in contexts where graphical elements play an important role in attracting users, such as choosing a mobile app from an app store. Therefore, visual aesthetics assessments are crucial in interface design, but traditional methods, involving target user representatives assessing each GUI individually, are costly and time-consuming. In this context, machine learning models have been demonstrated to be promising in automating the assessment of GUIs of web-based software systems. Yet, solutions for the assessment of mobile GUIs using machine learning are still unknown. Here we introduce a deep learning model to assess the visual aesthetics of mobile Android applications designed with App Inventor. We used a supervised learning approach to train and compare models using three different architectures. The highest performing model, a Resnet50, achieved a mean squared error of .022. The assessments of new GUIs showed an excellent correlation with human ratings ($\rho = .9$), and the Bland Altman plot analysis revealed 95% agreement with their labels. These results indicate the model's effectiveness in automating the visual aesthetics assessment of GUIs of mobile apps.

Keywords: Aesthetics; Mobile Application; Android; Deep Learning; Automatic Assessment

1 Introduction

Visual aesthetics is one essential factor in perceived usability, interaction, and overall appraisal of graphical user interfaces (GUIs) [Hamborg *et al.*, 2014; Tuch *et al.*, 2012]. It represents an integral part of usability as a GUI quality and refers to its beauty or pleasing appearance [ISO, 2011; Tractinsky *et al.*, 2000]. Positive aesthetic responses often lead to positive interface interactions [Bhandari *et al.*, 2019] and give users the immediate and long-lasting impression that a software system is suitable and easy to use [Norman, 2002; Tractinsky *et al.*, 2006; Tuch *et al.*, 2012]. Appealing GUIs may still compensate for poor perceived usability [Anderson, 2011; Bhandari *et al.*, 2019; Zen and Vanderdonckt, 2016]. Furthermore, perceived visual aesthetics often determines if systems will be used or avoided in favor of competition [Lu *et al.*, 2014; Schenkman and Jönsson, 2000; Tractinsky, 2013]. A first impression of the visual design is often the differentiating factor that becomes decisive in choosing an app from the myriad of options available in the major app stores [Bhandari *et al.*, 2019; Miniukovich and De Angeli, 2015].

Although there is considerable research on different aspects of visual aesthetics on desktop or web interfaces, it has received limited attention on mobile GUIs [Bhandari *et al.*, 2017; Lima and Gresse von Wangenheim, 2021; Miniukovich and De Angeli, 2014b]. GUIs on mobile devices are essentially different from those on other devices [Punchoojit and Hongwarittorrn, 2017]. For example, mobile GUIs have changed the traditional interaction models based on the familiar WIMP (Windows, Icons, Menus, Pointer) interface style, presenting a novel paradigm with widgets, touch, physical motion, on-screen keyboards, and sensor information [Flora *et al.*, 2014]. Touch-sensitive mobile displays also offer a variety of movements, such as swiping, pinching, spreading, and flicking. And since fingertips are usually bigger than mouse arrows, touchable elements need to be large enough to avoid misselection. In addition, the size and portability requirements limit what can be displayed at once on the screen to avoid cluttering, extending it down to several screens, and requiring scrolling [Rahmat *et al.*, 2018]. Differences are also related to the usage context, as a broader spectrum of people with different goals typically use mobile apps anywhere, anytime [Huang, 2009]. Furthermore, mobile device users highly value visual aesthetics because on-the-go usage implies multiple external distractions for short but intensive interaction periods [Choi and Lee, 2012]. Those factors directly influence interaction and interface design on these devices, demanding even greater attention concerning interface development [Flora *et al.*, 2014].

The visual aesthetics of mobile GUIs must be adequately assessed given its importance [Moshagen and Thielsch, 2010]. A typical approach to assessing visual aesthetics is to ask target users to indicate their perception of GUIs' overall appearance [Lavie and Tractinsky, 2004; Moshagen and Thielsch, 2010]. However, this is an expensive and time-consuming method that demands considerable resources that might be unavailable to small companies or individual professionals [Miniukovich and De Angeli, 2015]. One way to minimize that effort is by automating assessments of the visual aesthetic of GUIs to quickly detect and visualize problematic design aspects [Miniukovich and De Angeli, 2014a]. Automatic assessments are helpful in application development, especially in the early stages of the process, because they demand less effort than traditional assessment methods. They can also benefit non-professional designers and developers, considering the significant trend in software technology where more and more people with a background in other domains are developing mobile applications to solve problems related to their areas [Paternò, 2013]. Software development by end-users has become possible through block-based visual programming languages [Wolber *et al.*, 2015]. For example, App Inventor is an intuitive block-based programming environment that allows everyone, even children, to create fully functional mobile Android apps for smartphones and tablets. App Inventor has been around for over 12 years, with over 14.9 million users worldwide [MIT App Inventor, 2022]. However, current automated solutions to assess applications created with App Inventor are mostly limited to evaluating computational thinking concepts [Alves *et al.*, 2019]. Approaches, including the automatic assessment of GUI aspects, are still scarce and only cover adherence to style guides [Solecki *et al.*, 2020]. An example is CodeMaster [Gresse von Wangenheim *et al.*, 2018a] for the assessment of the conformity of App Inventor GUIs with Material Design guidelines in the context of computing education in K-12 [Alves *et al.*, 2019]. Yet, no automated method that analyzes the visual aesthetics of mobile applications has been encountered so far.

In a more general context, diverse methods are applied to automate visual aesthetics assessments of different kinds of artifacts [Seckler *et al.*, 2015; Zen and Vanderdonckt, 2016]. These methods aim to analyze how interface features and layout elements influence users' perception of visual aesthetics [Kim *et al.*, 2003]. Approaches vary from the simple numerical count of interface elements to more complex algorithms that analyze handcrafted features, such as colorfulness and symmetry [Miniukovich and De Angeli, 2015; Seckler *et al.*, 2015; Zen and Vanderdonckt, 2016]. But, by examining design factors independently, these methods may not capture the full complexity of visual aesthetics perception of the whole GUI [Bhandari *et al.*, 2019; Dou *et al.*, 2019]. Furthermore, there is still no consensus on how to consistently assess GUI visual aesthetics [Lima and Gresse von Wangenheim, 2021; Zen and Vanderdonckt, 2016].

More recently, deep learning models have been applied to quantify the visual aesthetics of the design of GUIs of webpages [Dou *et al.*, 2019; Khani *et al.*, 2016; Xing *et al.*, 2021] following the success of the evaluation of the aesthetics of photographs [Deng *et al.*, 2017; Lu *et al.*, 2014; Malu *et al.*, 2017]. Deep learning approaches can automatically extract high-level features directly from raw input data to predict the visual aesthetics of images [LeCun *et al.*, 2015; Polyzotis *et al.*, 2017]. Most of these models adopt a supervised learning approach where they compare their predictions with the actual labels of the images used in the training phase [Li *et al.*, 2016]. That way, they can use a backpropagation algorithm to adjust their internal parameters to minimize the difference between them LeCun *et al.* [2015]. Those models typically represent visual aesthetics with discrete values, such as "ugly" or "beautiful," or as a numerical value ranging from [0..1], treating the assessment as either a classification or a regression task [Kirchner *et al.*, 2015]. Again, research focusing more specifically on mobile applications is almost non-existent [Lima and Gresse von Wangenheim, 2021].

Therefore, we propose the deep learning model Appsthetics to quantify the visual aesthetics of GUIs of Android apps created with App Inventor by adopting a regression-based supervised approach with convolutional neural networks. The main contributions of our research are:

- A labeled dataset with 820 App Inventor GUI screenshots[1];
- A deep learning model to automatically quantify the visual aesthetics of GUIs of Android apps with predicted aesthetics ratings that strongly correlate with human user rating data[2].

The article is organized as follows. Section 2 reviews related work. Section 3 presents the methodology followed for the model development. Section 4 describes the development process, starting with how the model was designed, following through the steps taken to build the dataset and to train it, up to the results produced by two training rounds. The model evaluation, with the correlation and the Bland-Altman analysis, is in section 5. The discussion about those results and how they compare with human aesthetic perception is in section 6. Finally, section 7 concludes the paper and brings suggestions for future work. This work is an extended and revised version of an article already published in the IHC 2022 [Lima *et al.*, 2022c].

2 Related Work

Previous research on user experience, usability, or human-computer interaction, including GUI visual aesthetics, has focused on assessing web-based GUIs on large screens such as desktops or laptops [Lima and Gresse von Wangenheim, 2021]. The assessment of visual aesthetics on mobile GUIs has typically been extended from how GUIs are assessed on those screens. Much of this research has focused on handcrafted features. Miniukovich and De Angeli [2014b] validated six features for assessing web interfaces (color depth, dominant colors, visual clutter, symmetry, figure-ground contrast, and edge congestion) that can help predict the visual quality of Android [Miniukovich and De Angeli, 2014a] and iPhone GUIs [Miniukovich and De Angeli, 2015]. Other studies analyze the correlation between GUI complexity and user-perceived quality in Android applications, eventually proposing guidelines for GUI complexity by mining

[1] Available at https://bit.ly/app-inventor-dataset-v2.

[2] Available at https://bit.ly/appsthetics-code.

available Android applications [Alemerien and Magel, 2014; Taba *et al*., 2014].

With the advances in machine learning, deep learning approaches have been introduced to predict GUI visual aesthetics [Dou *et al*., 2019; Khani *et al*., 2016; Xing *et al*., 2021]. Some approaches employ pre-trained networks to reduce their training effort. Khani et al. [2016] trained their model based on the AlexNet architecture using a dataset of 418 website screenshots labeled using a rating according to their visual aesthetics. They used a hybrid model integrating convolutional neural networks (CNN) with classical machine learning techniques. The model uses a support vector machine to generate the end output with a Gaussian radial basis function to classify each image's visual aesthetics as "good" or "bad." As a result of the approach, they report an error rate (root mean squared error) of 34.15%.

Dou et al. [2019] also used a pre-trained CNN (CaffeNet) to train their model with 398 website screenshots. But unlike Khani et al., their dataset was labeled with the mean value of all human ratings received on a 9-point scale, with "1" meaning the lowest visual aesthetics and "9" the highest, rather than using only two categories. With the label values ranging from 1 to 9, predicting visual aesthetics was treated as a regression task. Their reported error rate is 20.41%.

Instead of using pre-trained networks, Xing et al. [2021] trained five distinct models with 38,423 GUI images collected from a popular website for UI designers in China. The dataset has been labeled using the number of "likes" the GUIs have received and the number of user "collections" to which they belong as their visual aesthetics representations. They achieved the best performance with a Squeeze-and-Excitation VGG model. Although an excellent error rate of 14.89% predicting "likes" and 25.38% predicting "collections" is reported, no proposal to unite these two individual results into one unique visual aesthetics value is presented.

More recently, Bakaev et. al [Bakaev *et al*., 2022] trained artificial neural networks (ANNs), that generally need fewer data, and CNN models based on a modified GoogLeNet architecture [Szegedy *et al*., 2015] for the assessment of visual aesthetics, complexity, and orderliness of about 2,700 web GUI from different domains. Whereas the CNN models used the GUI screenshots as inputs, the ANN models received 32 normalized metric scores obtained for the screenshots. When comparing those models, the ANN performed better than the CNN, achieving a MSE = .772 (87.86% error) against a MSE = .968 (98.38% error).

Although the reported error rates in the existing research seem high, they may be regarded as acceptable because no other results had been previously published and may serve as reference points. Dou et al. [2019] suggest treating the task of predicting visual aesthetics as a regression problem rather than a classification problem, attributing the performance improvements obtained compared to Khani et al. [2016] mainly to the training model used and the change to regression. Despite the success of convolutional neural networks predicting the aesthetic values of photographs, webpages, or GUI designs, no research on the application of CNNs to assess mobile GUI visual aesthetics has been found [Lima and Gresse von Wangenheim, 2021].

3 Research Methodology

Aiming at automating visual aesthetics assessments of App Inventor GUIs, we developed a deep learning model following the machine learning process proposed by Ashmore et al. [2021] and Amershi et al. [2019].

Requirements analysis. Based on the related work about different types of GUIs identified through a literature review, we defined the main objective of the model and the specification of its target features, following Mitchell [1997]. This step also includes the characterization of the input and the expected outputs, specifying the problem.

Data management. This step includes selecting available generic datasets for the model pre-training and collecting GUI screenshots to build the domain-specific dataset. After the screenshot collection, we cleaned the dataset by removing duplicates. Adopting a supervised learning technique, each screenshot was labeled corresponding to its visual aesthetics based on human ratings. To ensure that the labels represent the visual aesthetics perception of the target users, a central tendency score was computed from the individual rates that a group of volunteers manually assigned to the GUI screenshots. We then pre-processed the dataset, resizing collected screenshots to fit the deep learning architecture before training. It was then split into a training set to train the model and a validation set to perform an unbiased performance evaluation of the chosen model on unseen data. We also reserved 20 screenshots for testing [Ripley, 2007].

Training and performance evaluation. Based on the literature, we chose an appropriate deep learning framework and model that has proven effective for this problem type, volume, and data structure [He *et al*., 2016; Simonyan and Zisserman, 2015; Tan and Le, 2020]. We adopted a supervised transfer learning approach, which allowed us to start training from a pre-trained network with a generic dataset, accelerating the process and reducing the dependence on a large number of target-domain data [Iman *et al*., 2023; Zhuang *et al*., 2021]. That is possible because instead of using an untrained deep learning network, with random weight for its nodes, we began by selecting a network that had been pre-trained with a generic but related dataset, reducing learning costs [Iman *et al*., 2023]. The training is executed until the network no longer improves its performance. After transfer learning, we unfroze the internal features, allowing all network layers to learn. Then, the model went through a new training phase, called fine-tuning, with the same domain-specific dataset, to finely adjust all internal features to the data. A set of hyperparameters (momentum and learning rates) for the learning algorithm is dynamically selected and optimized during the fine-tuning process. We trained some similar model variants with different learning rates to compare and select those with the best results.

We defined the evaluation metric in alignment with the goals to be achieved and the type of problem faced to evaluate its performance. Performance is measured with the validation set through the defined metric, allowing it to analyze its result and make adjustments aiming at its improvement. Then the model is tested against previously unseen data (test set).

Model evaluation. To know to what degree the model predictions are equivalent to the human assessments, we executed a correlation analysis between the values resulting from the model and those attributed by the human evaluators. Furthermore, the correlation analysis allows comparing the results of this model with those of other works that perform the same evaluation method. We also analyzed the Bland & Altman agreement [Bland and Altman, 1986] to measure the degree of agreement between the two assessment methods (by the model and by humans) of the visual aesthetics of GUIs.

4 Appsthetics: A Deep Learning Model for Assessing the visual aesthetics of mobile apps

To support the automated assessment of the visual aesthetics of Android GUIs, we developed a CNN called Appsthetics. A summary of the development process is in Figure 1.

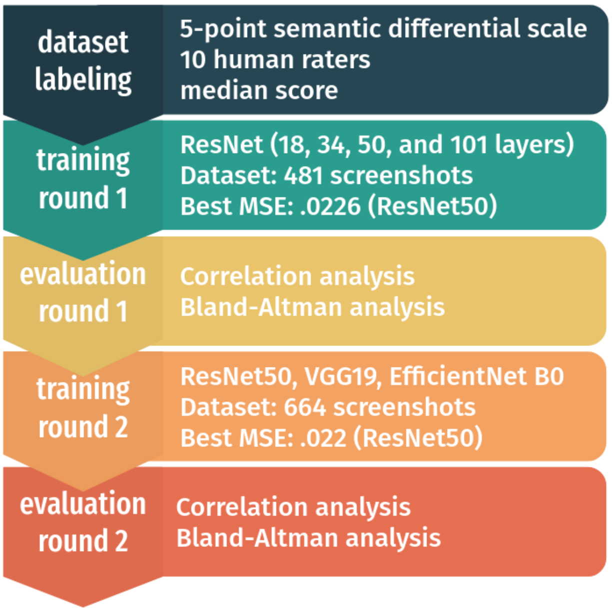


Figure 1. Summary of the Appsthetics development process.

4.1 Requirements Analysis

Our objective is to develop a deep learning model that learns from experience E for some class of tasks T and performance measure P, in which its performance on tasks in T, as measured by P, improves with E. In this research:

- A task in T is the assessment of the visual aesthetics of an App Inventor app screenshot with a numerical value within the interval [0..1];
- Experience E is a labeled dataset of App Inventor app screenshots, where each label is its degree of visual aesthetics within the interval [0..1]; and
- Performance P is the model loss, measured as the mean squared error (MSE) between the predicted visual aesthetics degrees and the actual screenshot labels. As for the model performance, we aim to achieve a testing error (MSE) below .03.

The model inputs are screenshots of Android applications developed with App Inventor and manually labeled by human raters. We adopted a supervised machine learning approach to deal with the visual aesthetics assessments and propose the aesthetics assessment task as a real-valued regression problem, rather than classification, following Dou et al. 2019. Our goal is to predict continuous scores for the screenshot visual aesthetics instead of discrete category labels. The first reason for our choice is that images are not either beautiful or ugly (or any other category between these two) but provoke aesthetic experiences in different degrees from person to person. Second, regression has achieved better results than classification in similar works [Dou *et al.*, 2019]. That way, the output is a numerical value within [0..1], interpreted as the visual aesthetics degree, where 0 = "very ugly" and 1 = "very beautiful."

We selected a high-level CNN framework called fast.ai to develop our machine learning model [Howard and Gugger, 2020]. It is based upon the PyTorch/Torch Python CNN framework, a good performing, flexible, and research-oriented CNN framework [Fonnegra *et al.*, 2017]. It offers ready-to-use and customizable functions to train models, making it suitable for practitioners mainly interested in applying pre-existing deep learning methods [Howard and Gugger, 2020]. Our choice for fast.ai relies on the fact that we are performing research on interface design assessments employing deep learning techniques rather than advancing state-of-the-art deep learning technologies.

We executed two training rounds to be sure we were using the best architecture for our problem. In the first round, we trained models using residual network architectures (ResNets), including ResNet18, ResNet34, ResNet50, and ResNet101 [He *et al.*, 2016], with a dataset of 481 screenshots. In the second round, we expanded the dataset with 183 new screenshots and trained the best-performing architecture in the previous round (ResNet50) and two other architectures, a VGG19 [Simonyan and Zisserman, 2015] and an EfficientNet B0 [Tan and Le, 2020], to compare their performances. Until recently, these architectures have been among those with the best performance in image recognition tasks [Dosovitskiy *et al.*, 2021; Xing *et al.*, 2022].

ResNets employ identity connections that act as shortcuts, bypassing several layers at once, providing two parallel learning paths in several network sections, and avoiding the typical gradient loss of very-deep networks. This architecture allows for much deeper networks with up to 152 layers. Due to their design principle, we could choose or design a network with adequate depth for the complexity of the problem at hand. We chose ResNets mainly because they enable hyperparameter optimization strategies (HYPOs) especially developed for these CNN models, allowing faster training [Smith, 2018; Smith and Topin, 2019].

The VGG19 architecture has achieved high accuracy with large-scale image recognition [Simonyan and Zisserman, 2015] and significant results with the visual aesthetics assessments of photographs [Lin, 2022; Sakaguchi *et al.*, 2022]. It uses small 3x3 convolution filters, the smallest possible size that still captures up/down and left/right. All hidden layers use ReLU as the activation function. The EfficientNet B0 uses a fixed set of coefficients to scale up width, depth, and image resolution uniformly Tan and Le [2020] and overcome the difficulty of randomly scaling up each of those dimensions by trial and error. The result is performance improvement with less use of computational resources. We selected EfficientNet B0, which has shown a performance similar to ResNet50 [Tan and Le, 2020].

4.2 Data Management

To build the dataset, we captured screenshots from apps available in the MIT App Inventor Gallery and from apps developed in the context of the Software Quality Group of the Universidade Federal de Santa Catarina (GQS/INE/UFSC). The dataset preparation included eliminating duplicates and images with unacceptable content (commercial, political, religious, or anti-ethical). We also pre-processed the screenshots to reduce and standardize the screenshot sizes (downsampling).

Data Collection. Six members of our research group participated in the process following a predefined script. As many apps available in the App Inventor gallery result from class assignments in K-12, some apps are incomplete or very rudimentary. Therefore, only apps with at least one visible component were selected, eliminating blank or unfinished screens.

After loading each application project into the App Inventor IDE, we used its live test feature to run the GUIs. All screenshots were captured using Genymotion v.3.1.2 to emulate a Google Pixel device running Android 8.0 - API 26 and were saved as PNG images with 1080x1920 pixel resolution. The process resulted in 8,303 screenshots from 1,552 different applications. We selected 820 App Inventor screenshots from this pool to create the dataset. The following criteria were applied, excluding:

- screenshots in landscape mode;
- screenshots displaying text in alphabets other than the Latin alphabet;
- screenshots of games, as their interface design significantly differs from other types of apps;
- screenshots of unfinished apps (e.g., blank screens and screens displaying only one element in the upper left corner or placeholder for text);
- screenshots very similar to others already selected (e.g., GUIs containing maps or variations of the same application); and
- screenshots displaying unacceptable content.

Furthermore, aiming at creating a balanced dataset, the screenshots were selected to include those ranging from very ugly to very beautiful, based on the authors' perception.

Data Labeling. Due to the lack of a consensus on classification scales to assess visual aesthetics as a one-dimensional construct in literature [Lima *et al.*, 2022a], we conducted an exploratory study to compare scale alternatives [Lima *et al.*, 2022b]. In this study, 208 subjects[3] rated the visual aesthetics of ten different mobile GUIs on Likert and semantic differential scales, with five and seven points each. Participants were randomly divided into four groups. Each group used a different rating scale to assess the GUIs. They also responded to the short version of the Visual Aesthetics of Websites Inventory (VisAWI-S) [Moshagen and Thielsch, 2013], a questionnaire widely used to assess the visual aesthetics of websites, considered a golden standard. Inter- and intra-examiner reliability and agreement were calculated for each scale and compared with the VisAWI-S to compute its validity.

The study results showed that all four scale types allow excellent inter-rater reliability, with an intraclass coefficient (ICC) above .98. The intra-rater agreement was also good, with Kendall's coefficient of concordance around Wt = .6. Although both Likert scales had the highest scores, they were only slightly higher than the semantic differential scale scores. Responses with 5-point scales resulted in lower intra-rater reliability than with 7-point ones, and the 5-point Likert scale resulted in the lowest intra-rater agreement score among the others. All four scales showed to be valid when compared to the VisAWI-S questionnaire and demonstrated a good correlation with each other when compared two by two (between .83 and .93). Based on these results, we chose the 5-point semantic differential scale to label the dataset.

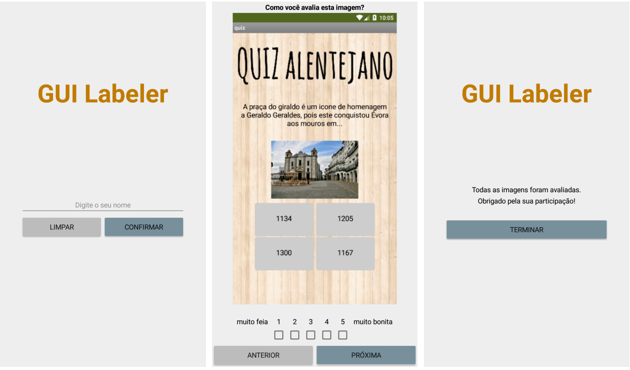
Ten members of our research group participated in the labeling process (50% female), all with degrees in computing-related areas. None of them reported being colorblind. Four participants reported having no experience with interface design, and another had less than one year of experience. The other five participants had at least two years of experience. Each participant used their own device to label the screenshots in three sessions. In the first two sessions, they labeled 600 screenshots that were used in the first training round (dataset 1). In the last session, they labeled another 220 screenshots that were added to the expanded dataset used in the second training round (dataset 2). Although three participants had iOS devices, only one was unfamiliar with Android applications.

Participants labeled the screenshots on a 5-point semantic differential scale (1 = "very ugly"; 5 = "very beautiful"). This scale type has shown high reliability and validity when used to rate mobile GUI visual aesthetics [Lima *et al.*, 2022b]. It also demonstrated a good correlation with the short version of the Visual Aesthetics of Website Inventory (VisAWI-S) [Moshagen and Thielsch, 2013], showing a corresponding validity with only one item, considerably reducing the rating effort.

We developed an application with App Inventor to operationalize the rating process. The app enabled participants to assign a degree of perceived visual aesthetics to each of the screenshots (**Figure 2**) during the labeling process. The app shows each screenshot separately with the respective rating scale below. The rating process can be interrupted at any moment, allowing its continuation from where it stopped. This allowed participants to halt their assessments whenever needed or wanted, e.g., due to an external interruption or fatigue. In addition, it enabled participants to rate the apps anywhere and anytime, bringing them closer to real-life users. The responses from this rating process showed excellent inter-rater reliability (ICC(C,10) = .877; 95% CI [.862; .891]).

To compute the final label of each screenshot, we cal-

[3] Approval for conducting this research with human participants was granted by the Human Research Ethics Committee (CEPSH) at UFSC (Certificate No. 4.971.708).



Como você avalia esta imagem?

A praça do giraldo é um icone de homenagem a Geraldo Geraldes, pois este conquistou Évora aos mouros em...

Todas as imagens foram avaliadas.
Obrigado pela sua participação!

Figure 2. Application developed for the rating process.

culated the median of all ratings as the indicated measure of central tendency for ordinal scales [Nunnally and Bernstein, 1994]. Besides, the median can provide a typical value that is not as skewed by extremely high or low scores. As each screenshot received an even number of ratings (10), the median could result in the intermediate value between two points on the rating scale. We decided to keep these values to minimize the loss of granularity from converting the scale values to the continuous interval [0..1]. Thus, although the screenshots received ratings on a 5-point scale, their labels could have nine possible values (five scale points and their intermediate values). We also computed the average absolute deviation (AAD) as a measure of dispersion among the responses [Leys *et al.*, 2013]. A high deviation indicates that the ratings are spread along the rating scale, allowing us to interpret that participants disagree about the visual aesthetics of that particular screenshot. Therefore, we removed 97 screenshots that received ratings with a deviation equal to or greater than 1 from dataset 1 to avoid training our model with confusing data. In addition, another 22 screenshots (all with a deviation of .9) were removed to balance the set between beautiful and ugly images. Applying the same criteria. 17 screenshots were removed from dataset 2. Finally, we normalized the labels to the interval [0..1], where 0 = "very ugly" and 1 = "very beautiful."

As a result, dataset 1 contained 481 GUI screenshots and dataset 2 contained 684 screenshots. All images had labels indicating their visual aesthetics degrees within [0..1]. We randomly set aside screenshots with average absolute deviations of .5 or less to form the test set, fairly distributed along the rating scale. The test set had 15 screenshots in the first training round, and 20 in the second round. None of these screenshots took part in the training or validation steps. We randomly divided the other screenshots using the proportion of 80% for training and 20% for validation. That way, dataset 1 was split into a training set of 373 screenshots and a validation set of 93 screenshots, and dataset 2 into a training set of 532 screenshots and a validation set of 132 screenshots. For pre-processing, we downsampled the screenshots from 1080x1920 to 448x448 pixels. We performed no other transformations, such as cropping, to avoid distorting image features relevant to visual aesthetics perception. The dataset is available online https://bit.ly/app-inventor-dataset-v2.

4.3 Model Training

To train our models, we used a pre-trained model with ImageNet, one of the largest publicly available general-purpose datasets [Russakovsky *et al.*, 2015]. Dealing with a regression problem, we adapted the input layer to the image resolution of our dataset, represented by the screenshot vector and its numeral label. The original output layers representing a categorical variable with 1,000 values to classify the ImageNet categories (containing 1,000 neurons) were replaced by a regression layer with a single neuron. Although the screenshots received ratings on an ordinal 5-point scale, we aim to predict continuous scores for the screenshot visual aesthetics instead of discrete category labels. That is because the cross-entropy loss function of classification models would not reflect the distances between different points on the rating scale. For example, "2" is closer to "3" than it is to "5," but the cross-entropy loss would be the same. Also, regression has achieved better results than classification in similar works [Dou *et al.*, 2019; Xing *et al.*, 2021]. That way, the output is a numerical value within [0..1], interpreted as the visual aesthetics degree, where 0 = "very ugly" and 1 = "very beautiful."

The performance quality of a deep learning model indicates how well its predictions match up against the ground truth [Botchkarev, 2019]. A typical quality measure for regression tasks is the mean squared error (MSE). MSE is the average of the squares of the errors. i.e., the average squared difference between the data labels (human rating scores) and the value of the deep learning model [Aggarwal, 2018]. MSE values are always non-negative values, with the lower, the better.

We executed two training rounds, each one with transfer learning and fine tuning phases. In the first round, we trained ResNets of varying depths (ResNet18, ResNet34, ResNet50, and ResNet101) with dataset 1 containing 466 screenshots. In the second round, we trained the best performing model of the previous round and to other architectures (VGG19 and EfficientNet B0) with dataset 2 (664 screenshots) to compare their performances. The results are presented below.

Round 1

The output layers of the networks were transfer-trained until the validation error stopped improving. We trained two models for each architecture, one using standard training (fit) and another using automated hyperparameter optimization (fit1cycle), with transfer-learning and fine-tuning phases [Smith, 2018; Smith and Topin, 2019]. This strategy works with a varying adaptive learning rate and momentum, where the learning rate is automatically increased first and then decreased while the momentum rate follows the opposite way [Smith, 2018]. We kept all default parameters and trained for no more than 100 epochs during transfer learning (see **Table 1**).

In the fine-tuning phase, we employed the same strategy as in the transfer learning, unfreezing and allowing for the adaptation of all weights in the network. We determined a range of optimum learning rates using the method suggested by Smith and Topin [2019]. It resulted in a different range of rates for each network. After fine-tuning, all models slightly

Table 1. Summary of the compared models in the first round

Architectures	ResNet18, ResNet34, ResNet50, ResNet101	
Input dimension	448 (pixels) x 448 (pixels) x 3 (color channels)	
Predictive model	Regression [0..1]	
Learning algorithm	Backpropagation	
Dataset separation	373 for training (80%); 93 for validation (20%)	
Training strategy	fit and fit1cycle	
Phase	Transfer learning	Fine tuning
Epochs	100	20
Learning rate	.003	range of LRs
Weight decay	No	No
Best MSE	.0306	.0227
Architecture	ResNet101	ResNet 50

improved their performance. Given the results presented in Table 2, ResNet50 trained with the fit strategy performed best (see **Figure 3**).

Table 2. Best MSE for each model in the first round

	Transfer learning		Fine tuning	
Architecture	Strategy	MSE	LR	MSE
ResNet18	fit	.0366	6.3e-07, 3.3e-07	.0359
	fit1cycle	.0344	4e-06, 1.3e-03	.0320
ResNet34	fit	.0310	3.3e-06, 9.1e-06	.0301
	fit1cycle	.0323	6.3e-07, 7.6e-08	.0323
ResNet50	fit	.0381	1e-04, 1e-03	.0226
	fit1cycle	.0345	1e-05, 1e-04	.0268
ResNet101	fit	.0306	3e-06, 3e-07	.0279
	fit1cycle	.0320	2.7e-06, 2.1e-04	.0271

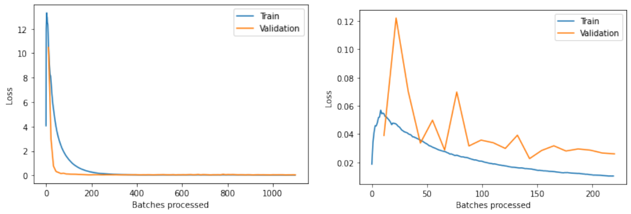


Figure 3. Train and validation losses for ResNet50 in the first round; transfer learning (left); fine-tuning (right)

These results demonstrate that the model performed very well in classifying visual aesthetics. Most classifications (81.7%) differed by less than one point when converted back to a 5-point scale. That means that most of the time, it can classify a "beautiful" GUI somewhere between "very beautiful" and "neither beautiful nor ugly" but never classifies such a GUI as "ugly," for example. That is an acceptable result, considering that even humans have trouble agreeing about visual aesthetics [Gresse von Wangenheim *et al.*, 2018b].

These good performance results have also been observed when using the test set, containing only unseen screenshots without access to their labels for prediction **Figure 5**. Considering only this set, the MSE was .0166 and for only two GUIs the prediction differed by one point when converted to a five-point scale. The model differed by just half a point on the visual aesthetics degree of eight GUIs and got the label right on another five.

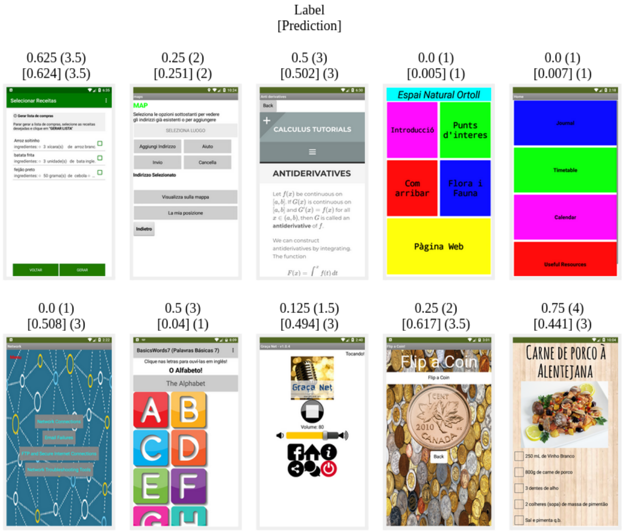


Figure 4. Validation set: best (top) and worst (bottom) classifications (scores converted to the 5-point scale in parenthesis)

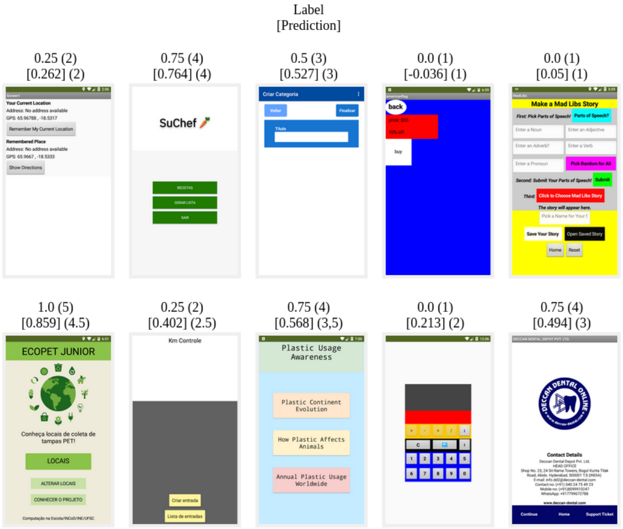


Figure 5. Test set: best (top) and worst (bottom) predictions (values converted to the 5-point scale in parenthesis)

Round 2

In this round we expanded the training/validation dataset with 198 new screenshots and randomly selected 20 screenshots for the test set. We trained a ResNet50, which was the best performing model in the previous round and a VGG19 and an EfficientNet B0. That way, we could compare the performance of the ResNet with that of two architectures that are widely used for similar tasks. Here, we used the same training steps (transfer learning and fine tuning) and strategies used in round 1 (see **Table 3**).

Again, the Resnet50 model presented the best performance when compared with the other architectures (see **Table 4**). Although the VGG19 and the EfficientNet B0 showed superior performance when compared with the other ResNets, in this round the ResNet50 showed lower MSE than in the previous round.

The ResNet50 model kept good performance when classi-

Table 3. Summary of the compared models in the first round

Architectures	VGG19, ResNet50, EfficientNet B0	
Input dimension	448 (pixels) x 448 (pixels) x 3 (color channels)	
Predictive model	Regression [0..1]	
Learning algorithm	Backpropagation	
Dataset separation	532 for training (80%); 132 for validation (20%)	
Training strategy	fit and fit1cycle	
Phase	Transfer learning	Fine tuning
Epochs	100	20
Learning rate	.003	range of LRs
Weight decay	No	No
Best MSE	.0238	.022
Architecture	EfficientNet B0	ResNet 50

Table 4. Best MSE for each model in the second round

	Transfer learning		Fine tuning	
Architecture	Strategy	MSE	LR	MSE
VGG19	fit	.027	6.3e-07, 6.9e-05	.0256
	fit1cycle	.03	6.3e-07, 5.7e-05	.0285
ResNet50	fit	.0268	1e-04, 1e-03	.022
	fit1cycle	.0293	1e-05, 1e-04	.0277
EfficientNet B0	fit	.024	1e-05, 1e-04	.0239
	fit1cycle	.0238	2e-04, 2e-05	.0236

fying visual aesthetics. In this round, 79.5% of the classifications differed by less than one point when converted back to a 5-point scale, which is very close to the previous round. On the other hand, the classifications for only five screenshots (out of 132) differed by one and a half points from their labels (**Figure 7**, bottom).

The model also had very good performance with the test set (**Figure 8**). When converted to a five-point scale, predictions differed by at most one point and were correct for eight out of twenty GUIs (40%).

5 Model Evaluation

We also evaluated the model to assess how close the visual aesthetics predicted in our models are to human ratings. That is typically done by conducting a correlation analysis [Purchase *et al.*, 2011; Miniukovich and De Angeli, 2015; Dou *et al.*, 2019], which we executed to enable the comparison of our results to a similar work [Dou *et al.*, 2019]. Although correlations quantify the degree to which two variables are related, they do not indicate how much they agree. As they only evaluate the linear association between two sets of observations, they can be inadequate and misleading when assessing their degree of agreement [Giavarina, 2015]. Thus, we also used the Bland-Altman (B&A) plot analysis to measure the degree of agreement between the automatic assessment of our model and the human ratings [Bland and Altman, 1986].

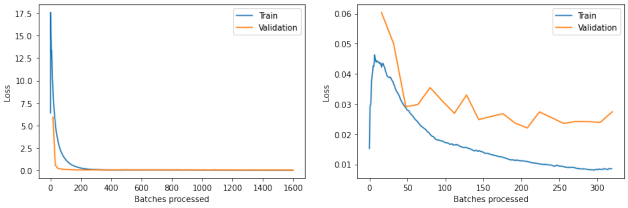


Figure 6. Train and validation losses for ResNet50 in the second round; transfer learning (left); fine-tuning (right)

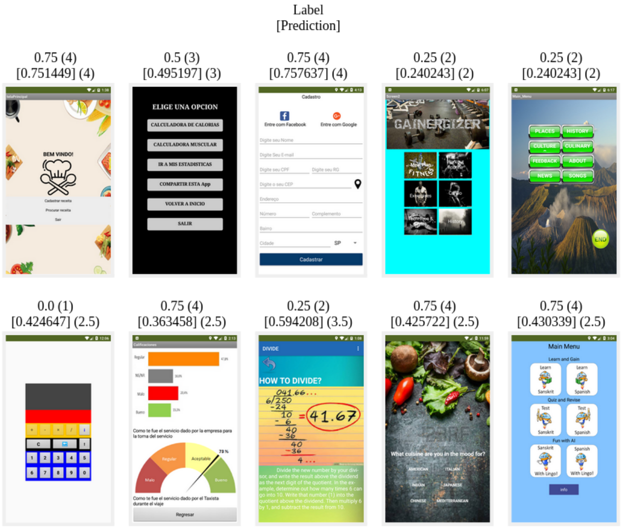


Figure 7. Validation set: best (top) and worst (bottom) classifications (scores converted to the 5-point scale in parenthesis)

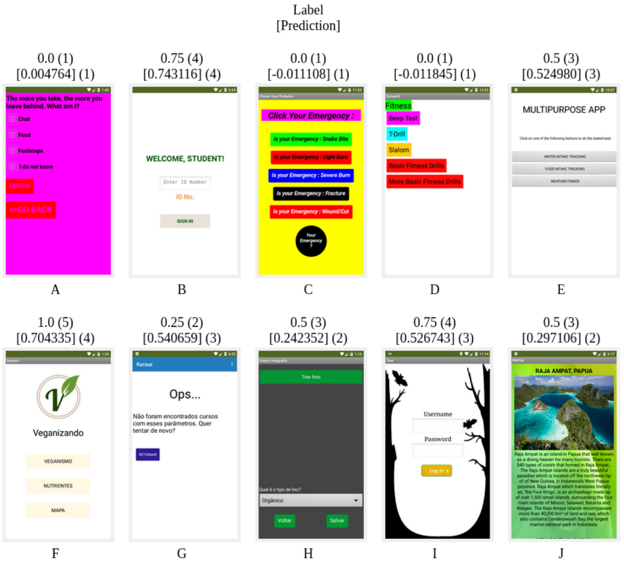


Figure 8. Test set: best (top) and worst (bottom) predictions (values converted to the 5-point scale in parenthesis)

5.1 Correlation Analysis

To evaluate if the trained model performs well on previously unseen inputs, we analyzed the performance of the learned model against the test dataset. We measured the strength of the linear association between the results of the deep learning network and ground truth based on human assessments using the Spearman rank correlation (ρ) [Bonett and Wright, 2000]. The choice for Spearman's rank correlation coefficient, rather than the more common correlation test Pearson's r, is justified by the non-normality of the data [Bryman and Cramer, 1990]. The numerical value of ρ ranges from -1 to $+1$. The closer the coefficients are to -1 or $+1$, the stronger the linear relationship is.

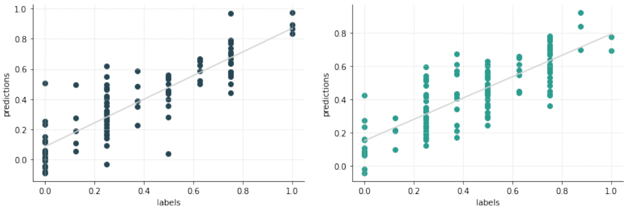


Figure 9. Correlation between labels and predictions in the first round (left) and in the second round (right) on the validation set

In the first round, the ResNet50 model trained with the fit strategy showed the best correlation between predictions and labels ($\rho = .87$) on the validation set (**Figure 9**, left). In the second round, that same model trained with a larger set (dataset 2) achieved a correlation of $\rho = .79$. When comparing the performance with a similar work, the correlation in the first round is on par with Dou et al. [2019], that report a correlation of $r = .85$ on the validation set. That work, however, used a dataset with a normal-like distribution, where most of the labels are close to the center of the scale and very few or no labels are close to its ends (**Figure 10**, left). Such a dataset can bias the model and improve the chance of getting the prediction right in the validation set, since it follows the same distribution. We tried to use datasets that are as balanced as possible (**Figure 10**, right). The difficulty was finding samples on the upper end of the scale ("very beautiful" GUIs). Nonetheless, except for these GUIs, the model learned to classify all others with the same chance.

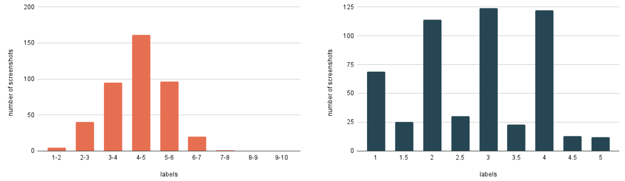


Figure 10. Distribution of samples on the validation set in Dou et al. [2019] (left) and in the second training round (right)

After both training rounds, our models performed very well predicting the visual aesthetics of the screenshots in the test set (**Figure 11**). The ResNet50 model trained with the fit strategy showed the correlation of $\rho = .95$ in the first round and $\rho = .9$ in the second one ($\rho = .9$). That is an excellent correlation, considering that the models sere assessing images that they had not seen before.

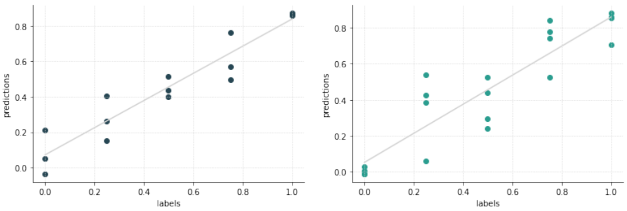


Figure 11. Correlation between labels and predictions in the first round (left) and in the second round (right) on the test set

We also correlated the predictions with the labels on the set containing the removed screenshots (average absolute deviations equal to or greater than 1) in both training rounds. All models resulted in ρ between .50 and .66, indicating that our choice to remove those screenshots about which humans show a higher degree of disagreement on their visual aesthetics was correct.

5.2 Bland-Altman Analysis

As correlation analysis shows the relationship between two variables, not their differences, we also performed a Bland-Altman (B&A) plot analysis to assess how they compare [Giavarina, 2015]. The B&A plot analysis describes the agreement between two quantitative measurements by studying the mean difference and constructing limits of agreement. It allows us to evaluate a bias between the mean differences and estimate an agreement interval, within which 95% of the differences between the first and the second methods fall [Bland and Altman, 1986]. This analysis does not indicate if the agreement between the predicted values and the human ratings is sufficient or if the automated assessment is suitable to replace the human one. It only quantifies the bias and a range of agreement, within which 95% of the differences between one measurement and the other are included [Giavarina, 2015]. B&A recommends that 95% of the data points lie within ±2 standard deviations of the mean difference.

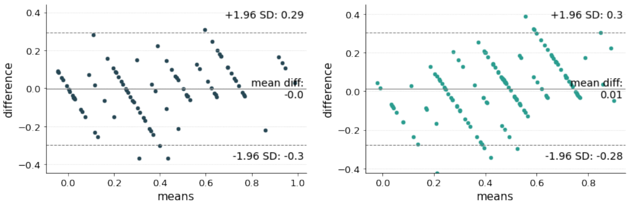


Figure 12. B&A plot analysis on the validation set in the first round (left) in the second round (right)

Figure 12 shows the B&A charts for the validation sets. It can be seen that the average difference between labels and predictions is zero in the validation set in the first and .01 in the second round. That is an indication of bias absence predicting visual aesthetics in the validation sets. The confidence interval (CI) is within the expected range. In the first round, the CI varied from -.3 to .29, and in the second round, it ranged from -.28 to .3, showing that 95% of the predictions differ from the labels by just over one point or less on a five-point scale.

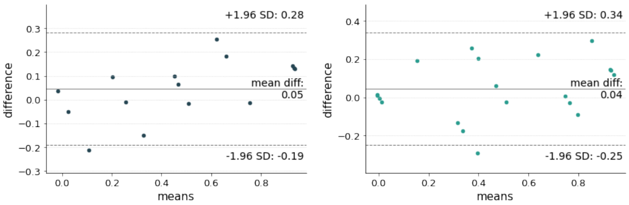


Figure 13. B&A plot analysis on the test set in the first round (left) in the second round (right)

For the test set, the labels are on average .04 larger than the predictions in the first round and .05 in the second one (see **Figure 13**). This is less than half a point on a 5-point scale. However, it also shows a slight tendency of the model to assign a lower degree of visual aesthetics than humans. The CI in the first round (-.19 to .28) was smaller than in the second one (-.25 to .34). It shows that 95% of the predictions differ from the labels by just over one point or less on a 5-point scale.

6 Discussion

Automating the assessment of the visual aesthetics of GUIs is challenging, as different people may rate it differently, and the optimal computational representation of aesthetics is far from obvious. And although there exist already first proposal approaches using deep learning to assess the visual aesthetics of web pages, we evolve the current state of the art by focusing specifically on mobile GUIs, with different characteristics than web pages.

Regarding performance, the Appsthetics obtained an MSE below .022, surpassing the assessment of web page GUIs (MSE = .042) [Dou *et al.*, 2019]. Our model also performed slightly better than the one assessing GUI designs (MSE = .0222) [Xing *et al.*, 2021] but predicted visual aesthetics directly, not indicators of user aesthetic preference. Still, we achieved that result with a much smaller dataset since we adopted transfer learning. We agree with Dou et al. [2019] that formulating the problem as a regression task is a significant factor for that performance. Previous informal tests with classification models also yielded lower performance results. The evaluation results demonstrate that a convolutional neural network can learn to predict the visual aesthetics of mobile GUIs based on their screenshots. The Appsthetics predictions showed an excellent correlation with the human ratings ($\rho = .9$), with the B&A plot analysis indicating that more than 95% of them agree, i.e., 19 out of the 20 outputs are within the 95% confidence interval. These results outperform other models assessing web GUIs [Dou *et al.*, 2019; Khani *et al.*, 2016] in an unprecedented approach for mobile GUIs.

However, the model seems to have trouble predicting the visual aesthetics of some GUIs on the higher end of the rating scale. As the B&A plot analysis indicates, the model tends to assign lower values to screenshots than humans. One reason might be that the training dataset is unbalanced, containing more "ugly" than "beautiful" screenshots. We detected that problem after the first training round and tried to mitigate it by adding more beautiful screenshots when composing dataset 2. Nonetheless, the human raters' preferences did not reflect the authors' and the new beautiful GUIs did not receive the highest ratings from the participants. Also, creating a more balanced dataset has been complicated because a large majority of the apps available in the App Inventor Gallery have rather ugly interface designs, making it difficult to encounter beautiful designs in larger quantities. Despite this bias, the B&A plot analysis also indicates that the difference is at most one point on the five-point scale for 95% of screenshots. That means that no GUI labeled as 4 ("beautiful") received a 2 ("ugly") from the model or vice versa.

It is also interesting to note the considerable difference in the correlations of the human assessments with the test set and with the one containing those GUIs removed from training due to the high mean absolute deviation in labeling. The test set consisted of GUIs on which human raters expressed a high degree of agreement as to their visual aesthetics. When assessing the screenshots from this group, the results showed a strong correlation with the human ratings ($\rho = .9$). The B&A analysis also indicated the agreement between the two assessment methods. On the other hand, the correlation between the model results and the set of GUIs excluded from the training was considerably lower ($\rho = .61$). These results show that just as humans have difficulty agreeing on the visual aesthetics of some GUIs, so does the deep learning model as expected. And although it represents an objective representation of visual aesthetics, it derives from the GUI properties and human intersubjectivity when rating them, composed of different subjective evaluations. Therefore, the deep learning model is susceptible to the same difficulties humans face when assessing GUIs with conflicting or confusing aesthetic elements.

Examining the evaluation examples presented in Figure 8, we can also try to understand which design elements contribute to the visual aesthetics of GUIs. Those that received the lowest ratings, i.e., GUIs considered "very ugly," make heavy use of very saturated colors (A, C, and D). Long pieces of text also seem to reduce the visual aesthetics ratings (G and J). On the other hand, GUIs that received intermediate ratings have large areas of blank space (E, G, and H). Some higher-rated GUIs also use whitespace, with an additional contrast between colors much softer than on ugly screens (B, F, and I). We also observe that a lack of symmetry between the elements seems to contribute to lower visual aesthetics (A, D, and G). Finally, we noticed that GUIs with fewer large elements (B and F) receive better evaluations than GUIs with many small ones (A, C, and D). However, representing just a superficial analysis of this issue, the automatization of visual aesthetics can also support such an analysis in detail on a larger scale with reasonable effort.

Threats to validity. A potential threat to our results study relates to using a dataset that does not represent the full spectrum of possible outcomes. To minimize this threat, we tried to balance the dataset concerning the aesthetic ratings. Nonetheless, a complete balance was not achieved due to the small number of App Inventor apps with more beautiful interfaces. Another threat comes from the subjective character of human classification during labeling. To reduce it, we analyzed the inter-rater agreement of the human responses and removed those screenshots about which the human raters disagreed on their visual aesthetics. A further threat concerns labeling many GUIs at once, which can be affected by tired raters. For that reason, we instructed raters to interrupt labeling whenever they felt fatigued to mitigate this threat. Also, based on related work, we chose well-tested CNN architectures that had been used for similar tasks. To reduce the threat of selecting a model with suboptimal performance, we trained models with different depths and architectures to compare their results. Because we had a considerably small dataset for training deep learning models, we started from pre-trained models and applied a transfer learning technique to reduce the risk of overfitting. For evaluation, we selected appropriate methods following related work and theory to evaluate correlation and agreement. Concerning external validity, we used a considerable sample size for evaluation, with a large variety of application types that allow the generalization of the results. The performance of the deep learning model was analyzed separately based on a test set that was not previously used for training or validation and that was randomly chosen from the dataset.

7 Conclusion

This article presents an innovative approach to automatically assess the visual aesthetics of the GUIs of mobile Android applications developed with App Inventor, adopting deep learning. To train the model, we built a dataset with 820 GUIs with visual aesthetics labels based on human raters' perceptions. We trained and compared three different architectures, with Resnet50 presenting the best performance results (MSE = .022). Results of the evaluation of the model show that it can effectively provide aesthetic predictions with high correlation and agreement with human assessments. The proposed model can be used for effective and efficient visual aesthetics assessments during the design of GUIs for Android applications, as well as provide feedback to students in the context of teaching interface design. It is also another example of how deep learning technology can support the development process in software engineering and interface design. As part of future work, we intend to expand the dataset to allow a more detailed analysis of the design factors that influence the mobile GUI visual aesthetics. We also plan to integrate the model with the online automatic assessment tool Codemaster, improving the GUI analysis of apps designed in the context of computing education in K-12. In a complementary research line, we have plans to understand how an automatic assessment tool can detect what could be changed in each mobile GUI to improve their visual aesthetics. For that, we expect to combine deep learning methods with element-based techniques to analyze the contribution of objective factors to the overall visual aesthetics.

Declarations

Funding

This study was financed in part by the Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brasil (CNPq) - Grant No.: 303674/2019-9.

Authors' Contributions

Adriano Lima: Software, Validation, Writing- Original and new article preparation. Christiane Gresse von Wangenheim: Conceptualization, Methodology, Supervision, Writing- Original and new article preparation. Osvaldo P. H. R. Martins: Conceptualization, Methodology, Data Curation, Software, Writing- Original draft preparation. Aldo von Wangenheim: Software, Supervision, Writing- Original draft preparation Adriano F. Borgatto: Validation, Writing- Reviewing and Editing: Jean C. R. Hauck: Software, Writing- Reviewing, and Editing.

Competing interests

The authors declare that they have no competing interests.

References

Aggarwal, C. C. (2018). *Neural Networks and Deep Learning: A Textbook*. Springer International Publishing, Cham. DOI: 10.1007/978-3-319-94463-0.

Alemerien, K. and Magel, K. (2014). GUIEvaluator: A Metric-tool for Evaluating the Complexity of Graphical User Interfaces. In *Proceedings of the Twenty-Sixth International Conference on Software Engineering & Knowledge Engineering*, pages 13–18, Vancouver, BC, Canada. Available at: https://www.academia.edu/66448057/GUIEvaluator_A_Metric_tool_for_Evaluating_the_Complexity_of_Graphical_User_Interfaces.

Alves, N. d. C., Gresse Von Wangenheim, C., and Hauck, J. C. R. (2019). Approaches to Assess Computational Thinking Competences Based on Code Analysis in K-12 Education: A Systematic Mapping Study. *Informatics in Education*, 18(1):17–39. Available at: https://eric.ed.gov/?id=EJ1212844.

Amershi, S., Begel, A., Bird, C., DeLine, R., Gall, H., Kamar, E., Nagappan, N., Nushi, B., and Zimmermann, T. (2019). Software Engineering for Machine Learning: A Case Study. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, pages 291–300. DOI: 10.1109/ICSE-SEIP.2019.00042.

Anderson, S. P. (2011). *Seductive Interaction Design: Creating Playful, Fun, and Effective User Experiences*. New Riders Pub, Berkeley, CA, 1st edition edition. Book.

Ashmore, R., Calinescu, R., and Paterson, C. (2021). Assuring the Machine Learning Lifecycle: Desiderata, Methods, and Challenges. *ACM Computing Surveys*, 54(5):111:1–111:39. DOI: 10.1145/3453444.

Bakaev, M., Heil, S., Chirkov, L., and Gaedke, M. (2022). Benchmarking Neural Networks-Based Approaches for Predicting Visual Perception of User Interfaces. In Degen, H. and Ntoa, S., editors, *Artificial Intelligence in HCI*, Lecture Notes in Computer Science, pages 217–231, Cham. Springer International Publishing. DOI: 10.1007/978-3-031-05643-$7_1$4.

Bhandari, U., Chang, K., and Neben, T. (2019). Understanding the Impact of Perceived Visual Aesthetics on User Evaluations: An Emotional perspective. *Information & Management*, 56(1):85–93. DOI: 10.1016/j.im.2018.07.003.

Bhandari, U., Neben, T., Chang, K., and Chua, W. Y. (2017). Effects of Interface Design Factors on Affective Responses and Quality Evaluations in Mobile Applications. *Computers in Human Behavior*, 72:525–534. Place: Netherlands Publisher: Elsevier Science. DOI: 10.1016/j.chb.2017.02.044.

Bland, J. M. and Altman, D. G. (1986). Statistical Methods for Assessing Agreement Between Two Methods of Clinical Measurement. *The Lancet*, 327(8476):307–310. DOI: 10.1016/S0140-6736(86)90837-8.

Bonett, D. G. and Wright, T. A. (2000). Sample size requirements for estimating pearson, kendall and spearman correlations. *Psychometrika*, 65(1):23–28. DOI: 10.1007/BF02294183.

Botchkarev, A. (2019). A New Typology Design of Performance Metrics to Measure Errors in Machine Learning Regression Algorithms. *Interdisciplinary Journal of Information, Knowledge, and Management*, 14:045–076. DOI: 10.28945/4184.

Bryman, A. and Cramer, D. (1990). *Quantitative Data Analysis for Social Scientists*. Quantitative data analysis for social scientists. Taylor & Francis/Routledge, Florence, KY, US. Pages: xiv, 290.

Choi, J. H. and Lee, H.-J. (2012). Facets of Simplicity for the Smartphone Interface: A Structural Model. *International Journal of Human-Computer Studies*, 70(2):129–142. DOI: 10.1016/j.ijhcs.2011.09.002.

Deng, Y., Loy, C. C., and Tang, X. (2017). Image Aesthetic Assessment: An experimental survey. *IEEE Signal Processing Magazine*, 34(4):80–106. Conference Name: IEEE Signal Processing Magazine. DOI: 10.1109/MSP.2017.2696576.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv:2010.11929 [cs]*. arXiv: 2010.11929. DOI: 10.48550/arXiv.2010.11929.

Dou, Q., Zheng, X. S., Sun, T., and Heng, P.-A. (2019). Webthetics: Quantifying Webpage Aesthetics with Deep Learning. *International Journal of Human-Computer Studies*, 124:56–66. DOI: 10.1016/j.ijhcs.2018.11.006.

Flora, H. K., Wang, X., and V.Chande, S. (2014). An Investigation into Mobile Application Development Processes: Challenges and Best Practices. *International Journal of Modern Education and Computer Science (IJMECS)*, 6(6):1. DOI: 10.5815/ijmecs.2014.06.01.

Fonnegra, R. D., Blair, B., and Díaz, G. M. (2017). Performance Comparison of Deep Learning Frameworks in Image Classification Problems Using Convolutional and Recurrent Networks. In *2017 IEEE Colombian Conference on Communications and Computing (COLCOM)*, pages 1–6. DOI: 10.1109/ColComCon.2017.8088219.

Giavarina, D. (2015). Understanding Bland Altman Analysis. *Biochemia Medica*, 25(2):141–151. DOI: 10.11613/BM.2015.015.

Gresse von Wangenheim, C., Hauck, J. C. R., Demetrio, M. F., Pelle, R., Alves, N. d. C., Azevedo, L. F., and Barbosa, H. (2018a). CodeMaster - Automatic Assessment and Grading of App Inventor and Snap! Programs. *Informatics in Education - An International Journal*, 17(1):117–150. Available at: https://www.ceeol.com/search/article-detail?id=645618.

Gresse von Wangenheim, C., Porto, J. V. A., Hauck, J. C. R., and Borgatto, A. F. (2018b). Do We Agree on User Interface Aesthetics of Android Apps? *arXiv*, pages 1–5. arXiv: 1812.09049. DOI: 10.48550/arXiv.1812.09049.

Hamborg, K.-C., Hülsmann, J., and Kaspar, K. (2014). The Interplay between Usability and Aesthetics: More Evidence for the "What Is Usable Is Beautiful" Notion. *Advances in Human-Computer Interaction*, 2014:e946239. Publisher: Hindawi. DOI: 10.1155/2014/946239.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep Residual Learning for Image Recognition. pages 770–778. Available at:https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html.

Howard, J. and Gugger, S. (2020). Fastai: A Layered API for Deep Learning. *Information*, 11(2):108. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/info11020108.

Huang, K.-Y. (2009). Challenges in Human-Computer Interaction Design for Mobile Devices. In *Proceedings of the World Congress on Engineering and Computer Science*, volume 1, pages 236–241, San Francisco, USA. Available at:https://iaeng.org/publication/WCECS2009/WCECS2009_pp236-241.pdf.

Iman, M., Arabnia, H. R., and Rasheed, K. (2023). A Review of Deep Transfer Learning and Recent Advancements. *Technologies*, 11(2):40. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/technologies11020040.

ISO (2011). ISO/IEC 25010:2011, Systems and Software Engineering — Systems and Software Quality Requirements and Evaluation (SQuaRE) — System and Software Quality Models. Available at:https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/03/57/35733.html.

Khani, M. G., Mazinani, M. R., Fayyaz, M., and Hoseini, M. (2016). A Novel Approach for Website Aesthetic Evaluation Based on Convolutional Neural Networks. In *Proceedings of the 2016 Second International Conference on Web Research (ICWR)*, pages 48–53. DOI: 10.1109/ICWR.2016.7498445.

Kim, J., Lee, J., and Choi, D. (2003). Designing Emotionally Evocative Homepages: An Empirical Study of the Quantitative Relations Between Design Factors and Emotional Dimensions. *International Journal of Human-Computer Studies*, 59(6):899–940. DOI: 10.1016/j.ijhcs.2003.06.002.

Kirchner, J., Heberle, A., and Löwe, W. (2015). Classification vs. Regression - Machine Learning Approaches for Service Recommendation Based on Measured Consumer Experiences. In *2015 IEEE World Congress on Services*, pages 278–285. ISSN: 2378-3818. DOI: 10.1109/SERVICES.2015.49.

Lavie, T. and Tractinsky, N. (2004). Assessing Dimensions of Perceived Visual Aesthetics of Web Sites. *International Journal of Human-Computer Studies*, 60(3):269–298. DOI: 10.1016/j.ijhcs.2003.09.002.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep Learning. *Nature*, 521(7553):436–444. DOI: 10.1038/nature14539.

Leys, C., Ley, C., Klein, O., Bernard, P., and Licata, L. (2013). Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. *Journal of Experimental Social Psychology*, 49(4):764–766. DOI: 10.1016/j.jesp.2013.03.013.

Li, X., Zhang, G., Li, K., and Zheng, W. (2016). Deep Learning and Its Parallelization. In Buyya, R., Calheiros, R. N., and Dastjerdi, A. V., editors, *Big Data: Principles and Paradigms*. Morgan Kaufmann. Google-Books-ID: MfOeCwAAQBAJ. DOI: https://doi.org/10.1016/B978-0-12-805394-2.00004-0.

Lima, A. L. d. S. and Gresse von Wangenheim, C. (2021). Assessing the Visual Esthetics of User Interfaces: A Ten-Year Systematic Mapping. *International Journal of Human–Computer Interaction*, pages 1–21. DOI:

10.1080/10447318.2021.1926118.

Lima, A. L. d. S., Gresse von Wangenheim, C., and Borgatto, A. F. (2022a). Assessment of Visual Aesthetics through Human Judgments: a Systematic Mapping. In *Proceedings of the 21st Brazilian Symposium on Human Factors in Computing Systems*, IHC '22, pages 1–14, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/3554364.3560902.

Lima, A. L. d. S., Gresse von Wangenheim, C., and Borgatto, A. F. (2022b). Comparing Scales for the Assessment of Visual Aesthetics of Mobile GUIs Through Human Judgments. *International Journal of Mobile Human Computer Interaction (IJMHCI)*, 14(1):1–28. Publisher: IGI Global. DOI: 10.4018/IJMHCI.313028.

Lima, A. L. d. S., Martins, O. P. H. R., von Wangenheim, C. G., von Wangenheim, A., Borgatto, A. F., and Hauck, J. C. R. (2022c). Automated Assessment of Visual aesthetics of Android User Interfaces with Deep Learning. In *Proceedings of the 21st Brazilian Symposium on Human Factors in Computing Systems*, IHC '22, pages 1–11, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/3554364.3559113.

Lin, R. (2022). Augmenting Image Aesthetic Assessment with Diverse Deep Features. In *2021 4th Artificial Intelligence and Cloud Computing Conference*, AICCC '21, pages 30–38, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/3508259.3508264.

Lu, X., Lin, Z., Jin, H., Yang, J., and Wang, J. Z. (2014). RAPID: Rating Pictorial Aesthetics Using Deep Learning. In *Proceedings of the 22nd ACM international conference on Multimedia*, MM '14, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/2647868.2654927.

Malu, G., Bapi, R. S., and Indurkhya, B. (2017). Learning Photography Aesthetics with Deep CNNs. *arXiv:1707.03981 [cs]*. arXiv: 1707.03981. DOI: 10.48550/arXiv.1707.03981.

Miniukovich, A. and De Angeli, A. (2014a). Quantification of Interface Visual Complexity. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*, AVI '14, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/2598153.2598173.

Miniukovich, A. and De Angeli, A. (2014b). Visual Impressions of Mobile App Interfaces. In *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational*, NordiCHI '14, pages 31–40, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/2639189.2641219.

Miniukovich, A. and De Angeli, A. (2015). Computation of Interface Aesthetics. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 1163–1172. Association for Computing Machinery, New York, NY, USA. DOI: 10.1145/2702123.2702575.

MIT App Inventor (2022). MIT App Inventor | Explore MIT App Inventor. Available at: http://appinventor.mit.edu/.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill Education, 1 edition. Book.

Moshagen, M. and Thielsch, M. (2010). Facets of Visual Aesthetics. *International Journal of Human-Computer Studies*, 68(10):689–709. DOI: 10.1016/j.ijhcs.2010.05.006.

Moshagen, M. and Thielsch, M. (2013). A Short Version of the Visual Aesthetics of Websites Inventory. *Behaviour & Information Technology*, 32(12):1305–1311. DOI: 10.1080/0144929X.2012.694910.

Norman, D. (2002). Emotion & Design: Attractive Things Work Better. *Interactions*, 9(4):36–42. DOI: 10.1145/543434.543435.

Nunnally, J. C. and Bernstein, I. H. (1994). *Psychometric Theory*. McGraw-Hill, New York, 3rd edition. Book.

Paternò, F. (2013). End User Development: Survey of an Emerging Field for Empowering People. *ISRN Software Engineering*, 2013:e532659. Publisher: Hindawi. DOI: 10.1155/2013/532659.

Polyzotis, N., Roy, S., Whang, S. E., and Zinkevich, M. (2017). Data Management Challenges in Production Machine Learning. In *Proceedings of the 2017 ACM International Conference on Management of Data*, SIGMOD '17, pages 1723–1726, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/3035918.3054782.

Punchoojit, L. and Hongwarittorrn, N. (2017). Usability Studies on Mobile User Interface Design Patterns: A Systematic Literature Review. *Advances in Human-Computer Interaction*, 2017:e6787504. Publisher: Hindawi. DOI: 10.1155/2017/6787504.

Purchase, H. C., Hamer, J., Jamieson, A., and Ryan, O. (2011). Investigating objective measures of web page aesthetics and usability. In *Proceedings of the Twelfth Australasian User Interface Conference - Volume 117*, AUIC '11, pages 19–28, AUS. Australian Computer Society, Inc. Available at: https://www.researchgate.net/profile/Helen_Purchase/publication/262274197_Investigating_objective_measures_of_web_page_aesthetics_and_usability/links/5770224b08ae842225aa454c/Investigating-objective-measures-of-web-page-aesthetics-and-usability.pdf.

Rahmat, H., Zulzalil, H., Ghani, A. A. A., and Kamaruddin, A. (2018). A Comprehensive Usability Model for Evaluating Smartphone Apps. *Advanced Science Letters*, 24(3):1633–1637. DOI: 10.1166/asl.2018.11125.

Ripley, B. D. (2007). *Pattern Recognition and Neural Networks*. Cambridge University Press. Book.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252. DOI: 10.1007/s11263-015-0816-y.

Sakaguchi, D., Takimoto, H., and Kanagawa, A. (2022). Study on relationship between composition and prediction of photo aesthetics using CNN. *Cogent Engineering*, 9(1):2107472. DOI: 10.1080/23311916.2022.2107472.

Schenkman, B. N. and Jönsson, F. U. (2000). Aesthetics and Preferences of Web Pages. *Behaviour & Information Technology*, 19(5):367–377. DOI:

10.1080/014492900750000063.

Seckler, M., Opwis, K., and Tuch, A. N. (2015). Linking Objective Design Factors with Subjective Aesthetics: An Experimental Study on How Structure and Color of Websites Affect the Facets of Users' Visual Aesthetic Perception. *Computers in Human Behavior*, 49:375–389. DOI: 10.1016/j.chb.2015.02.056.

Simonyan, K. and Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv:1409.1556 [cs].

Smith, L. N. (2018). A Disciplined Approach to Neural Network Hyper-parameters: Part 1 – Learning Rate, Batch Size, Momentum, and Weight Decay. *arXiv:1803.09820 [cs, stat]*. DOI: 10.48550/arXiv.1803.09820.

Smith, L. N. and Topin, N. (2019). Super-convergence: Very Fast Training of Neural Networks Using Large Learning Rates. In *Proceedings of the Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, volume 11006, page 1100612. International Society for Optics and Photonics. DOI: 10.1117/12.2520589.

Solecki, I., Porto, J., Alves, N. d. C., Gresse von Wangenheim, C., Hauck, J., and Borgatto, A. F. (2020). Automated Assessment of the Visual Design of Android Apps Developed with App Inventor. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, SIGCSE '20, pages 51–57, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/3328778.3366868.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going Deeper With Convolutions. pages 1–9. Available at: https://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Szegedy_Going_Deeper_With_2015_CVPR_paper.html.

Taba, S. E. S., Keivanloo, I., Zou, Y., Ng, J., and Ng, T. (2014). An Exploratory Study on the Relation between User Interface Complexity and the Perceived Quality. In Casteleyn, S., Rossi, G., and Winckler, M., editors, *Web Engineering*, Lecture Notes in Computer Science, pages 370–379, Cham. Springer International Publishing. DOI: 10.1007/978-3-319-08245-5$_2$2.

Tan, M. and Le, Q. V. (2020). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks.

Tractinsky, N. (2013). Visual Aesthetics. *The Encyclopedia of Human Interaction*. Available at: https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/visual-aesthetics.

Tractinsky, N., Cokhavi, A., Kirschenbaum, M., and Sharfi, T. (2006). Evaluating the Consistency of Immediate Aesthetic Perceptions of Web Pages. *International Journal of Human-Computer Studies*, 64(11):1071–1083. DOI: 10.1016/j.ijhcs.2006.06.009.

Tractinsky, N., Katz, A., and Ikar, D. (2000). What is Beautiful is Usable. *Interacting with Computers*, 13(2):127–145. DOI: 10.1016/S0953-5438(00)00031-X.

Tuch, A. N., Roth, S. P., Hornbæk, K., Opwis, K., and Bargas-Avila, J. A. (2012). Is Beautiful Really Usable? Toward Understanding the Relation Between Usability, Aesthetics, and Affect in HCI. *Computers in Human Behavior*, 28(5):1596–1607. DOI: 10.1016/j.chb.2012.03.024.

Wolber, D., Abelson, H., and Friedman, M. (2015). Democratizing Computing with App Inventor. *GetMobile: Mobile Computing and Communications*, 18(4):53–58. DOI: 10.1145/2721914.2721935.

Xing, B., Cao, H., Shi, L., Si, H., and Zhao, L. (2022). AI-driven user aesthetics preference prediction for UI layouts via deep convolutional neural networks. *Cognitive Computation and Systems*, 4(3):250–264. DOI: 10.1049/ccs2.12055.

Xing, B., Si, H., Chen, J., Ye, M., and Shi, L. (2021). Computational model for predicting user aesthetic preference for GUI using DCNNs. *CCF Transactions on Pervasive Computing and Interaction*, 3(2):147–169. DOI: 10.1007/s42486-021-00064-4.

Zen, M. and Vanderdonckt, J. (2016). Assessing User Interface Aesthetics based on the Inter-subjectivity of Judgment. In *Proceedings of the 30th International BCS Human Computer Interaction Conference*, Poole, UK. BCS Learning & Development. DOI: 10.14236/ewic/HCI2016.25.

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., and He, Q. (2021). A Comprehensive Survey on Transfer Learning. *Proceedings of the IEEE*, 109(1):43–76. Conference Name: Proceedings of the IEEE. DOI: 10.1109/JPROC.2020.3004555.