FaceLift: A transparent deep learning framework recreating the urban spaces people intuitively love

ABSTRACT

In computer vision, deep learning techniques have recently been used to predict whether urban scenes are likely to be considered beautiful, and it turns out that these techniques do so quite accurately. To support urban interventions, however, one needs to go beyond *predicting* beauty and tackle the challenge of *recreating* beauty. Unfortunately, deep learning techniques have not been designed with that challenge in mind. Given their "black-box nature", they cannot even explain why a scene has been predicted to be beautiful. To partly fix that, we propose a deep learning framework (which we named FaceLift) that is able to both *beautify* existing Google Street views and *explain* which urban elements make those transformed scenes beautiful. To quantitatively evaluate our framework, we cannot resort to any existing metric (as the research problem at hand has never been faced before) and need to formulate new ones. These new metrics should ideally capture the presence (or absence) of elements that make urban spaces great. Upon a review of the urban planning literature, we identify five main metrics: walkability, green, landmarks, openness, and visual complexity. For all the five metrics, the beautified scenes are far better than the original ones. These results suggest that, in the future, as our framework's components are further researched and become better and more sophisticated, it is not hard to imagine technologies that will be able to accurately and efficiently support architects and planners in the design of the spaces we intuitively love.

ACM Reference format:
. 2017. FaceLift: A transparent deep learning framework recreating the urban spaces people intuitively love. In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 9 pages.
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Whether a street is considered beautiful is subjective, yet research has shown that there are specific urban elements that are universally considered beautiful: from greenery, to small streets, to memorable spaces [1, 16, 18]. These elements are those that contribute to the creation of what the urban sociologist Jane Jacobs called 'urban vitality' [11].

Given that, it comes as no surprise that computer vision techniques can automatically analyze pictures of urban scenes and accurately determine the extent to which these scenes are, *on average*, considered beautiful. Deep learning has greatly contributed to increase these techniques' accuracy [7].

Urban planners and architects are interested in urban interventions and, as such, wish to go beyond technologies that are only able to predict beauty scores. They often called for technologies that make their job of recreating beauty more accurate and efficient [3]. Deep learning is not fit for purpose. It is not meant to recreate beautiful scenes, not least because it cannot provide provide any explanation on why a scene is beautiful.

To partly fix that, we propose a deep learning framework (which we named FaceLift) that is able to both *generate* a beautiful scene (or, better, *beautify* an existing real urban scene) and *explain* why that scene is beautiful. In so doing, we make two main contributions:

- We propose a deep learning framework that is able to learn whether Google Street views are beautiful or not and, based on that training, is able to both *beautify* existing views (using state-of-the-art Generative Adversarial Networks) and *explain* which urban elements in these views make them beautiful (Section 3).
- We quantitatively evaluate whether the framework is able to actually produce beautified scenes (Section 4). We do so by proposing a family of five urban design metrics that we have formulated based on a through review of the literature in urban planning. For all these five metrics, the framework passes with flying colors: with minimal interventions, beautified scenes are twice as walkable as the original scenes. Also, after building an interactive tool with "FaceLifted" scenes and showing it to twenty experts in architecture, we found that the majority of them agreed on three main areas of impact: decision making, participatory urbanism, and promotion of restorative spaces among the general public.

We conclude by pointing out some limitations that might well guide future work (Section 5).

2 RELATED WORK

Previous work has focused on collecting ground truth data about how people perceive urban spaces, on predicting urban qualities (including beauty) from visual data, and on generating synthetic images that enhance a given quality (e.g., beauty).

Ground truth of urban perceptions. So far the most detailed studies of perceptions of urban environments and their visual appearance have relied on personal interviews and observation of city streets: for example, some researchers relied on annotations of video recordings by experts [19], while others have used participant ratings of simulated (rather than existing) street scenes [13]. The web has recently been used to survey a large number of individuals. Place Pulse is a website that asks a series of binary perception questions (such as 'Which place looks safer [between the two]?') across a large

Conference'17, Washington, DC, USA

DOI: 10.1145/nnnnnnn.nnnnnnn

number of geo-tagged images [18]. In a similar way, Quercia *et al.* collected pairwise judgments about the extent to which urban scenes are considered quiet, beautiful and happy [16]. They were then able to analyse the scenes together with their ratings using image-processing tools, and found that the amount of greenery in any given scene was associated with all three attributes and that cars and fortress-like buildings were associated with sadness. Taken all together, the results pointed in the same direction: urban elements that hinder social interactions were undesirable, while elements that increase interactions were the ones that should be integrated by urban planners to retrofit cities for greater happiness.

Deep learning and the city. Computer vision techniques have increasingly become more sophisticated. Deep learning techniques have been used to accurately predict urban beauty [7, 20], urban change [14], and even crime [4].

Generative models. Deep learning has recently been used not only to analyze existing images but also to generate new ones. Ngyuen *et al.* [15] used generative networks to create natural-looking image that maximizes a specific neuron. In theory, the resulting image is the one that "best activates" the neuron under consideration (e.g., that associated with urban beauty). In practice, it is still a synthetic image that needs further processing to look realistic.

To sum up, a lot of work has gone into collecting ground truth data about how people tend to perceive urban spaces, and into building accurate predictions models of urban qualities. However, little work has gone into models that generate realistic urban scenes enhancing desirable qualities and that offer human-interpretable explanations of what they generate.

3 FACELIFT FRAMEWORK

We present here Facelift, an end-to-end framework for image beautification. The framework embeds a model trained on a set of urban images annotated with beauty scores. It takes as input a geolocated urban image and gives as output its transformed (beautified) version. Although we refer here to specific urban properties (i.e. beauty) and datasets, Facelift is generalizable to any labeled dataset of geolocated images.

For the sake of berevity, we summarise the notations used in Table 1 and the framework steps in Figure 1.

In general terms, the framework allows anyone with an arbitrary set of *geolocated* images $X = I_1, I_2 ....I_n$ annotated in classes $Y = y_1, y_2, ..., y_k$, to transform natural images between classes: the algorithm can transform an image I_i belonging to class $y_i \in Y$, to image I_j from class $y_j \in Y$. Both I_i and I_j are natural, non-synthetic images. Despite having another *meaning* (i.e. category), I_j maintains the structural characteristics of I_i (e.g. point of view, layout). This allows to visually reason about the discriminative properties between classes $y_i, y_j \in Y$, and visually understand the salient characteristics that drive a classifier to distinguish between classes y_i, y_j.

The transformation framework consists of three phases (see Fig. 1). In the first phase, we classify images from X into the corresponding categories Y with high accuracy , using a convolutional neural network C. In our case, y_i and y_j are the beautiful/ugly classes. In the second phase, we transform am image from class y_i to class y_j, using Generative Adversarial Networks[17]. The output of this phase is a synthetic image $\hat{I}_j$, which summarizes the basic traits of the destination class $y_j \in Y$. The last phase matches the synthetic image $\hat{I}_j$, with the closest natural image in X. Finally, to quantitatively reason about the beautification process, we perform aggregated analysis of the differences between original images and resulting target images.We do this by quantifying the presence of 5 urban design metrics in uglified and beautified images.

For the rest of this section, we would delve deeper into the specifics of the image beautification framework.

symbol	**stands for**
X	Georeferenced urban image dataset
I_i	Georeferenced image $\in X$
Y	Annotations classes for X (e.g. beautiful/ugly)
y_i	Class in Y (e.g. beautiful)
$\hat{I}_j$	Template image
I'	Target Image
C	Image Classifier
R	Images acquired by rotating street view camera
T	Images acquired by translating street view camera
ρ	Similarity bound below which smart augmentation chooses translated images
term	**stands for**
Template Image $\hat{I}_j$	A synthetic transformation of input image I towards the class y_j
Target Image I'	The natural image which is most visually similar to the template image
Data Augmentation	A process of data expansion which looks for images taken in the surroundings of the georeferenced images in X
Classifier	A deep-learning framework that is able to classify images into one of the classes in Y
Generator (*GAN*)	A deep-learning based image generator
DGN − AM	A framework that, given the GAN and the Classifier, transforms an input image into the template image.

Table 1: Notations and Terms.

Step 1 Classifying Beauty

We design here a classifier C able to correctly assess the beauty category y_i of an image in X using a deep learning network. To reliably train a convolutional neural netowrk we need first make sure we have enough reliable data to train the classifier **[REF]**. We do this by augmenting the available geolocated image data.

Dataset and Beauty Judgments. Our seed dataset comes from Place pulse, a research work on urban affective dimensions [7]. The dataset in total contains 100k images across 56 cities around the world from Google StreetView[1]. Images are annotated through pairwise comparison for qualities such as beauty, depression, richness , safety etc. For the purpose of our work, we use the beauty judgements. To train our classifier C to detect beauty categories Y, we need to transform pairwise votes into absolute scores, then discretize absolute scores into a finite set of categories y_i. We transform the pairwise votes into ordinal scores using the TrueSkill [10] algorithm. To ensure reliability of absolute judgements, we filter out images with less than 3 votes. To discretize the resulting scores, we heuristically partition the data into two classes with maximum separation: beautiful and ugly. Figure 2 shows the distribution of Trueskill score

[1] https://maps.googleapis.com/maps/api/streetview

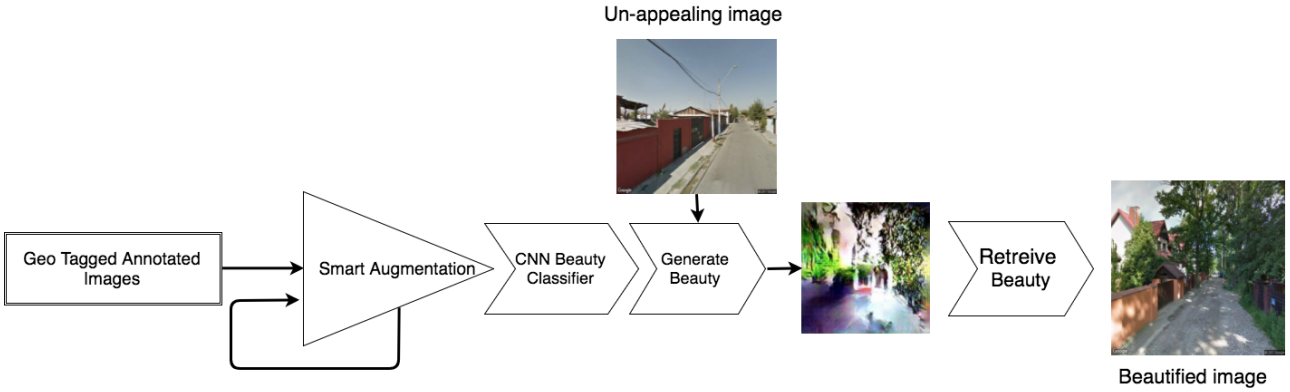


Figure 1: Architecture of the Beautification framework

estimates with the threshold scores at which we decide beauty or ugly class boundary.

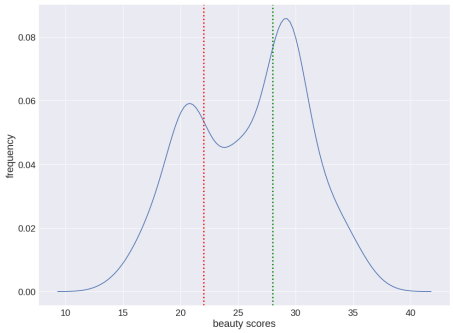


Figure 2: Distribution of ordinal scores for images with at-least 4 votes. The red and green line represent the threshold below and above which images are tagged Ugly or beautiful. Images in between are dropped for separability

Data Augmentation similarity bound. Despite over a hundred thousand images in the original data, only 20,000 has more than 3 judgements. This is non-ideal to train classifiers with substantial number of parameters such as convolutional neural network, since smaller data size implies that a machine learning model has a risk of over-fitting **[REF]**. We choose to augment the dataset by exploiting the geo-located nature of the image dataset. We also take advantage of the fact that urban places in close proximity look quite similar to each other [5] To develop a better way of augmenting images which can be transferred with scores of the original annotated ones, we make one heuristic assumption : "**[A1]***the composition of a StreetView image does not change considerably for small rotations of the camera angle*". This assumption was tried and tested over several samples both manually and using image similarity measures. An example of one such sample can be seen in Figure 3a. This assumption allows us to do a basic expansion of our dataset without adding a lot of noise. However we cannot to a similar assumption when it comes to translation of camera. All images in the PlacePulse dataset are taken with a default camera rotation which depends on the location. The Streetview API allows to specify the preferred camera rotation angle. This gives us the opportunity to take snapshots of the same location at different camera angles. We rotate the camera across different values $\theta \in -30°, -15°, 15°, 30°$ and derive a first set $R = \{R(I)_\theta \forall I \in X\}$, which consists of images acquired by rotating the camera angle for each annotated image in X. Following **A1**, For these images, the beauty score of each rotated image $R(I)_\theta$ can be safely transferred from I.

Next, for each image in X, we translate the location of the Streetview camera: we select points on the map at a distance of $d \in \{10, 20, 40, 60\}$ meters and acquire the resulting set of images $T = \{T(I)_d \forall I \in X\}$. Although possibly very similar, transferring the beauty score from I to each $T(I)_d$ might result in very noisy data. To understand the extent to which beauty scores can be transferred from images to their translated version, we use a smart augmentation technique.

In a nutshell, this technique computes the similarity between the translated and the original image, and transfers the beauty scores only if the similarity is acceptably high. To do so, we represent each images from both sets using visual features extracted from the FC7 layer of PlacesNet [23]. We then calculate the cosine similarities $S_t = \{s(I, T(I)_d) \forall I \in X\}$ between each original image I and all images in the augmented set $T(I)_d$. We also calculate another set of cosine similarities $S_r = \{s(I, R(I)_\theta) \forall I \in X\}$ between rotated and origianl images. We define a similarity bound as the median similarity betwen rotated and original images.

$$\rho = median(S_r) \text{ where } S_r = \{s(I, R(I)_\theta) \forall I \in X\} \quad (1)$$

Following the assumption **[A1]** , we only transfer beauty scores to translated images who look as similar to the original as their rotated counterparts: $s(I, T(I)_d) < \rho$. We discard translated images not fullfilling this requirement and retain the resulting images in the smartly translated image set $\hat{T}$.

Semantics of Augmentable images. Given that we now had a similarity bound to decide whether to augment or not a particular image, we wondered whether certain types of scenes are more prone to augmentation compared to others. So we partitioned our data in two sets

- $setA$: contains images where the median similarity between translated images and the original image $s(I, T(I)_d) < \rho$.

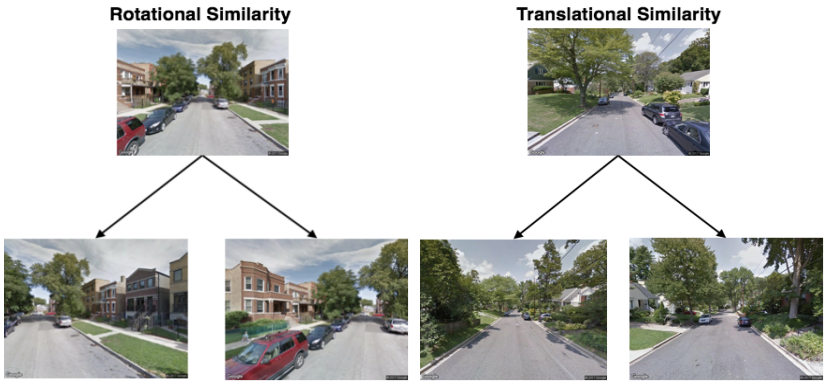


Figure 3: Fig 3a shows an example set of images showing similarity of streetview scapes, when the camera is rotated by a small angle. Fig 3b shows the translational similarity where the angle is less than the established bound ρ

- $setB$: Images whose similarity with their translated set is farther apart i.e. $s(I, T(I)_d) > \rho$.

We describe each image in both sets according to the scene depicted, by collecting the PlacesNet [23] labels with the top5 confidence scores. We then aggregate such labels at a set level by computing a TF-IDF metric. The resulting set of {label,Count} pairs reflect essentially how common or uncommon is a particular scene label in $setA$ compared to $setB$

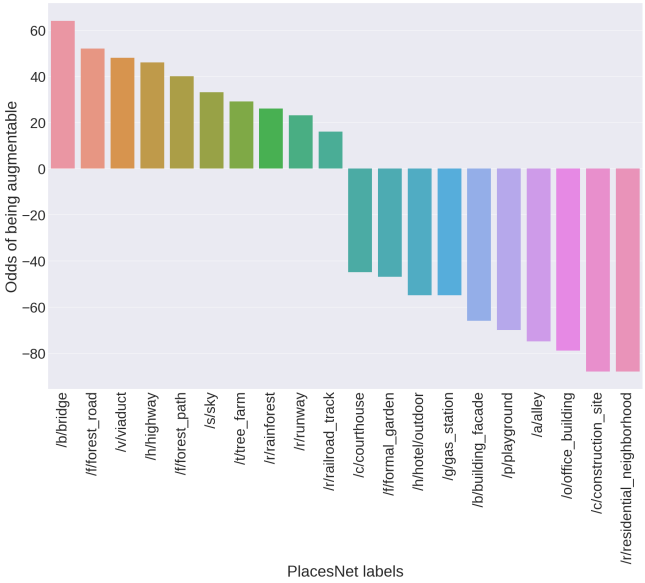


Figure 4: Prevalence plot of types of scenes prevalent in Similar images compared to dissimilar ones.

The resulting prevalences of scene types can be seen in Fig 4. The plot shows that scenes like highways, fields and bridges, typically more uniform and open, don't change despite translation. Other urban scenes like gardens, residential neighbourhoods , plazas and skyscrapers are more sensitive to change in perspective by translation.

The Beauty Classifier. Once we have enough data, we train a deep convolutional network to classify images into the Y beauty classes. One may use several successful deep convolutional neural network architectures, which work for other use cases like AlexNet [12] , PlacesNet [23] or GoogLeNet [21]. For our paper we use CaffeNet which is a modified version of AlexNet. This trained classifier is a important component in the next phase, which is generation of images.

We employ the above observations to train the beauty classifier model. We start with the base dataset of 20k images (X), as described in Section 3. We then progressively augment the data: first with rotation across 5 angles (X,R), then rotation with uniform translation for all images (X,R,T), and then rotation and translation for only images which satisfy similarity bound as evaluated as shown in Equation 1 (X,R,$\hat{T}$). We then train a Convolutional neural net model based on AlexNet architecture [21] on each of these augmentated datasets. The training is done on 70% split of the data and tested on the 30%.

We see considerable improvement in classifier accuracy 2, with the best model performing at 73% accuracy for classifying images in two classes of Beauty and Ugly. This model represents the concept of beauty, learned from annotated and augmented streetview images.

Policy	Accuracy (Percentage)
No augmentation (X)	63
Rotation only	68
Rotation + translation	64
Rotation + Smart Translation	73.5

Table 2: Performance differences based on different augmentation policies

Step 2 Generating Images

We now want to design a framework to transform any image I_i of class y_i into template image $\hat{I}_j$ (as shown in Figure 1), namely a synthetic version of the original image, with added features and motifs that maximize class y_j. To produce the template image $\hat{I}_j$, we need the following components in place:

- *Classifier*. We need a deep classifier C able to classify I into Y, i.e. between ugly and beautiful images.

- *Generator*. We train a generative adversarial network (GAN) which can generate an approximate natural looking image drawn from distribution of a particular class of images, similar to the one in [6].
- *Activation Maximization*. We plug in the GAN and the classifier network into an Activation Maximisation (AM) framework. Given these components, an input image I, and a target beauty class y_i, the AM transforms I in an ideal image $\hat{I}_j$ (that maximizes the activation for beauty class y_i).

We have described the design and performace of Classifier in Sec. **??**. We will delve deeper into the other two below

Generator. Generative Adverserial networks are an extremly useful tool when it comes to generating samples from a learned distribution[17]. GANs consist of a pair of networks where the *generator* generates image samples similar to an input space using de-convolutional layers, and the *discriminator* learns to discriminate between natural images from the training set and synthetic images generated by the generator. Since GANs are known to be very tricky to train [9] we first try to use a pre-trained GANs on Imagenet from [15].However, because of the vast difference between images in Imagenet and Streetview images, our initial results were not very optimal. We therefore retrained the generator on the StreetScore dataset. This improved the visual quality of the generated images considerably.

Activation Maximization. We build on top of the Activation Maximization technique elaborated by Nguyen et. al [15] (DGN-AM). DGN-AM utilizes the property of locality of codes:Generator codes which are close to each other would create similar looking images. DGN-AM was initially built to visualise the concept learnt by CNNs, by finding the code which maximised the activation of a particular output class in the classifier network. The maximization is achieved by doing gradient descent on the input generator codes with respect to the classifier neuronal activation, keeping everything else locked. The result is a synthetic image that has a high activation for a predetermined output neuron. We modify this method by starting the maximization method from a code K which corresponds to the a-priori input image, for example, an ugly urban image I_i. So for a given image I_i which belongs to class y_i (which could be the beauty neuron or the ugly neuron of the classifier C), the DGN-AM algorithm would perform Stochastic gradient descent on the generator codes of the a-priori image I_i so as to maximize the target neuron y_j (which could be beauty of ugly neuron of C resulting in a synthetic image $\hat{I}_j$ generated by the generator G from the code $\hat{K}$. The whole optimization can be expressed as Equation 2.

$$\hat{I}_j = G(\hat{K}) : \arg\max_{\hat{K}}(C_j(G(\hat{K})) - \lambda||\hat{K}||) \tag{2}$$

Here C_j corresponds to the activation of the neuron j of the classifier C , and G is the generator network. λ is the L_2 regularization factor. The resulting output image $\hat{I}_j$ is a natural-like image, which maximizes the beauty neuron for our classifier. We hypothesize that because the process begins from an a-priori image, the resulting image is closest possible template to the ugly input image, but with the beauty neural activation maximized. Figure 6 shows the activation maximization output in the center.

Step 3 Retrieving Images

In this final step we find a target image I' from the dataset that is closely aligned, in terms of some visual similarity metric $E(I_1, I_2)$, with the generated template image $\hat{I}_j$. The result of this exercise is to find the most similar looking image to an input image I that maximizes a particular annotation class y_j. The problem of finding images which are visually similar can be solved using image embeddings in a N dimensional space R^N We use a pre-trained deep network, which is trained to classify scene types to a very high accuracy [23] to extract the image embeddings. We extract a 4096 dimensional feature vector from the FC7 layer of the network to describe the the template image. We then extract feature vectors from the complete test dataset using the same process. We can now use the L_2 Norm to find pairwise distances in the R^{4096}. Formally with N test natural images and a template image $\hat{i}$ we extract $v_{\hat{i}} \in R^{4096}$ and and find pairwise distances $\{d_j \; \forall j \in N\}$ where $d_j = L_2(v_j, v_{\hat{i}})$ We then find the target image by finding the $min(\{d_j\})$. For the sake of redundancy, we find the top 5 such matches for every template $\hat{i}$ generated from every ugly image i. These target images are what we call the transformed images.

4 EVALUATION

The goal of our framework is to transform existing urban scenes into versions that: *i)* people perceive more beautiful; *ii)* contain urban elements typical of great urban spaces; *iii)* are easy to interpret; and *iv)* architects and urban planners find useful. To ascertain whether the framework meets that goal, next, we answer the following questions:

Q1 Do individuals perceive our framework's scenes to actually be beautified?
Q2 Does our framework produce scenes that possess urban elements typical of great spaces?
Q3 Which urban elements are mostly associated with beautiful scenes?
Q4 Do architects and urban planners find our framework useful?

Q1 People's perceptions of beautified scenes

To ascertain whether our framework's transformations are perceived by individuals as they are supposed to, we run a crowd-sourcing experiment on Amazon Mechanical Turk. We randomly select 200 scenes, 100 beautiful and 100 ugly based on whether they are at the bottom 10 and top 10 percentiles of the Trueskill's score distribution (Figure 2). Our framework then transforms each ugly scene into its beautified version, and each beautiful scene into its 'uglified'. These scenes are arranged into pairs, each of which contains a beautiful scene and an ugly one. On Mechanical Turk, we only select verified masters for our crowd-sourcing workers (those with an approval rate above 90% during the past 30 days), pay them $0.1 per task, and ask each of them to choose the beautiful scene in a pair. We make sure to have at least 3 votes for each scene pair. Overall, our workers select the scenes that are actually beautiful 77.5% of the times, suggesting that our framework's transformations are actually perceived by people to effectively be more beautiful.

Figure 5: Comparison of using the Default ImageNet GAN against Custom trained GAN for Activation maximization. By re-training the GAN on the test dataset, we can see improvement in terms of structure and colours in the generated images

Figure 6: Example of Beautification Process

Q2 Are beautified scenes great urban spaces?

To answer that question, we need to understand what makes a space great. After a careful review of the urban planning literature, we identify four factors [1, 8] (summarized in Table 3): great places mainly tend to be walkable, offer greenery, feel cozy, and be visually rich.

To automatically extract visual cues related to these four factors, we select 500 ugly scenes and 500 beautiful ones at random, transform them into their opposite aesthetic qualities (i.e., ugly ones are beautified, and beautiful ones are 'uglified'), and compare which urban elements related to the four factors distinguish uglified scenes from beautified ones.

We extract labels from each of our 1,000 scenes using two computer vision algorithms. First, using PlacesNet [23], we label our scenes according to a classification containing 205 labels (reflecting, for example, landmarks, natural elements), and retain the five labels with highest confidence scores for the scene. Second, using Segnet [2], we label our scenes according to a classification containing 12 labels. Segnet is trained on dash-cam images, and the resulting lables are road, sky, trees, buildings, poles, signage, pedestrians, vehicles, bicycles, pavement, fences, and road markings.

H1 Beautified scenes tend to be walkable. We manually select only the labels that are related to walkability. These labels include, for example, *abbey , plaza , courtyard, garden, picnic area, park*. To test hypothesis *H1*, we count the number of walkability-related labels found in beautified scenes as opposed to those found in uglified scenes (Figure 7): the former contain twice as much walkability labels that the latter. We then consider the prevalence of specific labels Figure 8, and find that beautified scenes tend to show gardens, yards, small path. By contrast, uglified ones tend to show built environment features such as shop fronts and broad sidewalks.

H2 Beautified scenes tend to offer green spaces. We manually select only the PlacesNet's labels that are related to greenery. These labels include, for example, *fields, pasture , forest, ocean, beach*. Then, in our 1,000 scenes, to test hypothesis *H2*, we count the number of nature-related labels found in beautified scenes as opposed to those found in uglified scenes (Figure 7): the former contain more than twice as much nature-related labels that the latter. To test this hypothesis further, we compute the fraction of 'tree' pixels (one of SegNet's label) in beautified and uglified scenes. We find that beautification adds 32% of greenery pixels, and uglification removes 17% of greenery pixels.

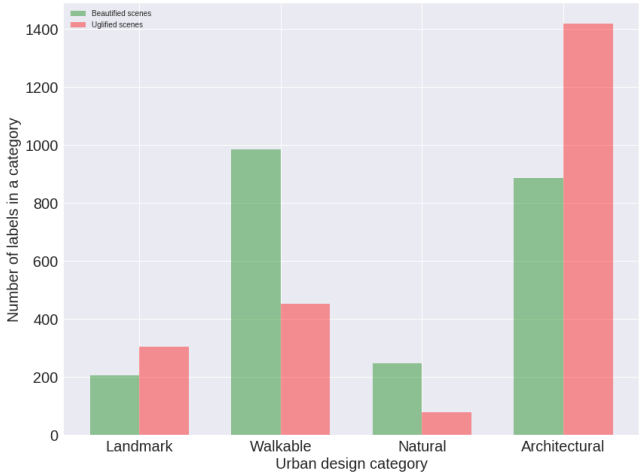


Figure 7: Number of labels in specific urban design categories (on the x-axis) found in beautified scenes as opposed to those found in uglified scenes.

H3 Beautified scenes tend to feel private and 'cozy'. To test hypothesis *H3*, we count the fraction of pixels that Segnet labeled as 'sky' and show the results in a bin plot in Figure 9a: the x-axis has six

Metric	Description
Walkability	Walkable streets are rated high on an aesthetic scale [8]. Walkable streets increase the social capital of a place and appeal to the exploring nature of human psyche. This implies that the urban space needs to address the fundamental need of people to walk and explore. This also implies that a walkable street must also be perceived as safe.
Green Spaces	Presence of Greenery is always pleasing to the eye. The literature always links urban beauty to curated and well maintained green spaces, where social interactions can happen [1]. This 'social' aspect of greenery implies that dense forests or unkempt greens are not always related to the sense of beauty in urban scenes.
Landmarks	Loosing a bearing in the city is not a very pleasant experience. Hence presence of recognisable and guiding landmarks influences the perception of an urban space [8].
Privacy-Openness	A sense of privacy and a complimentary sense of openness are both influential in our perception of a place[8]. These values also tend to be related in an inverse 'U' fashion with beauty.
Visual Complexity	Visual complexity is a measure of how diverse a urban scene is. It manifests in terms of various design materials, textures and objects. Generally, visual complexity has an inverse 'U' relation with aesthetic values. The beauty and aesthetics of a place increases until it starts dropping because of 'too much' complexity[8].

Table 3: Urban Design Metrics

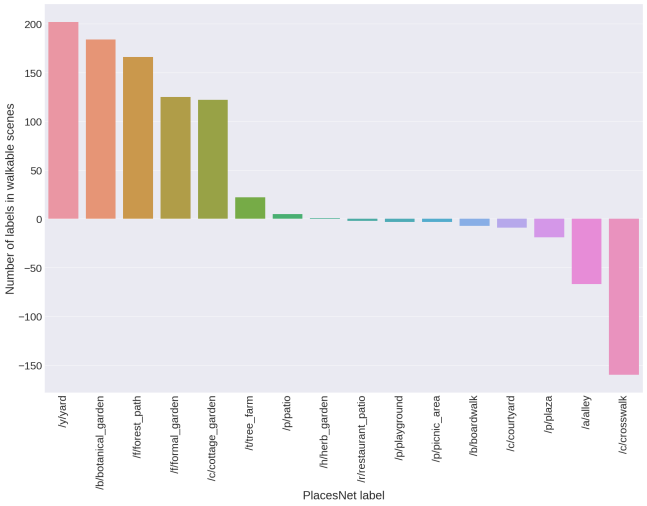


Figure 8: Count of specific walkability-related labels (on the x-axis) found in beautified scenes minus the count of the same labels found in uglified scenes.

bins each of which represents a given range of sky fraction, and the y-axis shows the percentage of beautified *vs.* uglified scenes that fall into the corresponding bin. Beautified scenes tend to be cozier (lower sky presence) than the corresponding original scenes.

H4 Beautified scenes tend to be visually rich. To quantify to which extent scenes are visually rich, we measure their visual complexity [8] as the amount of disorder in terms of distribution of (Segnet) urban elements in the scene:

$$H(X) = -\sum p(i) \log p(i) \quad (3)$$

where i is the i^{th} Segnet's label. The total number of labels is twelve. The higher $H(X)$, the higher the scene's entropy, that is, the higher the scene's complexity. To test hypothesis *H4*, we show the

Pair of urban elements	β_1	β_2	β_3	Error Rate (Percentage)
Buildings - Trees	-0.032	0.084	0.005	12.7
Sky - Buildings	-0.08	-0.11	0.064	14.4
Roads - Vehicles	-0.015	-0.05	0.023	40.6
Sky - Trees	0.03	0.11	-0.012	12.8
Roads - Trees	0.04	0.10	-0.031	13.5
Roads - Buildings	-0.05	-0.097	0.04	20.2

Table 4: Regression coefficients on a logistic run on a pair of predictors at the time.

percentage of scenes that fall into a given bin of complexity scores (Figure 9b): beautified images tend have low to medium complexity, while uglified scenes are of high complexity.

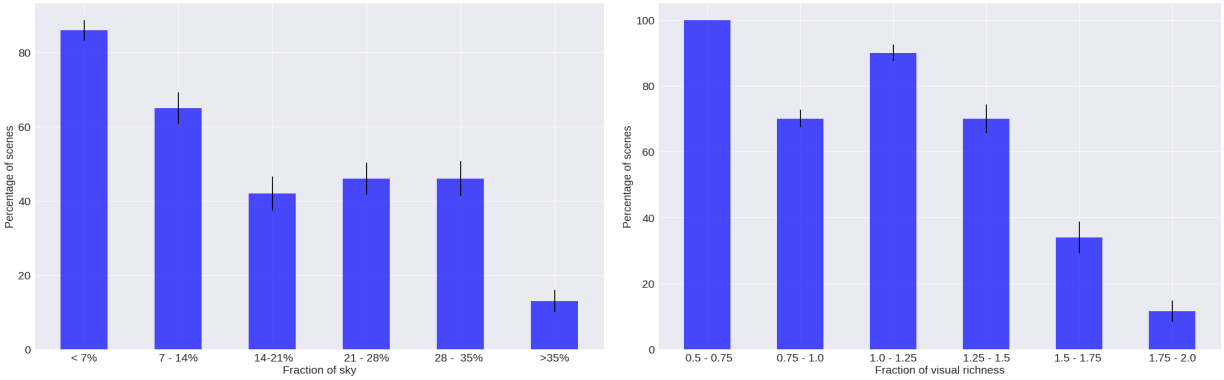
Q3 Urban elements of beautified scenes

To determine which urban elements are the best predictors of urban beauty and the extent to which they are so, we run a logistic regression, and, to ease interpretation, we do so on a pair of predictors at the time:

$$Pr(beautiful) = logit^{-1}(\alpha + \beta_1 * V_1 + \beta_2 * V_2 + \beta_3 * V_1.V_2) \quad (4)$$

where $V1$ is the fraction of the scene's pixels marked with one Segnet's label, say, buildings (over the total number of pixels), and $V2$ is the fraction of the scene's pixels marked with another label, say, trees. The result consists of three beta coefficients: β_1 reflects $V1$'s contribution in predicting beauty, β_2 reflects $V2$'s contribution, and β_3 is the interaction effect, that, reflects the contribution of the dependency of $V1$ and $V2$ in predicting beauty. We run logistic regression on the five factors that have been found to be most predictive of urban beauty [1, 8, 16].

Since we are using a logistic regression, the quantitative interpretation of the beta coefficients is eased by the "divide by 4 rule" [22]: we can take β coefficients and "divide them by 4 to get an upper bound of the predictive difference corresponding to a unit difference" in beauty [22]. For example, take the results in the first row of Table 4. In the model $Pr(beautiful) = logit^{-1}(\alpha - 0.032 \cdot buildings + 0.084 \cdot trees + 0.005 \cdot buildings \cdot trees)$, we can divide - 0.032/4 to get -0.008: a difference of 1 in the fraction of pixels being buildings



(a) **(b)**

Figure 9: The percentage of scenes (y-axis): (a) having an increasing presence of sky (on the x-axis); and (b) having an increasing level of visual richness (on the x-axis).

corresponds to no more than a 0.8% negative difference in the probability of the scene being beautiful. In a similar way, a difference of 1 in the fraction of pixels being trees corresponds to no more than a 0.021% positive difference in the probability of the scene being beautiful. By considering the remaining results in Table 4, we find that, across all pairwise comparisons, trees is the most positive element associated with beauty, while roads and buildings are the most negative elements. Since these results go in the direction one would expect, one might conclude that the scenes beautified by our framework are in line with previous literature, adding further external validity to our work.

Q4 Do architects and urban planners find it useful?

To ascertain whether practitioners find FaceLift potentially useful, we built an interactive map of the city of Boston in which, for selected points, we showed pairs of urban scenes before/after beautification **[picture?]**. We then sent that map along with a survey to 20 experts around the world in the areas of architecture, urban planning, and data visualization. The experts had to complete tasks in which they rated FaceLift based on how well it supports decision making, participatory urbanism, and promotion of green spaces among the general public. The results are show in Figure 10: according to our experts, the tool can very probably supports decision making, probably support participatory urbanism, and definitely promote green spaces. These results are qualitatively supported by our experts' comments, which included: "*The maps reveal patterns that might not otherwise be apparent*", "*The tool helps focusing on parameters to identify beauty in the city while exploring it*", and "*The metrics are nice. It made me think more about beautiful places needing a combination of criteria, rather than a high score on one or two dimensions. It made me realize that these criteria are probably spatially correlated*".

5 CONCLUSION

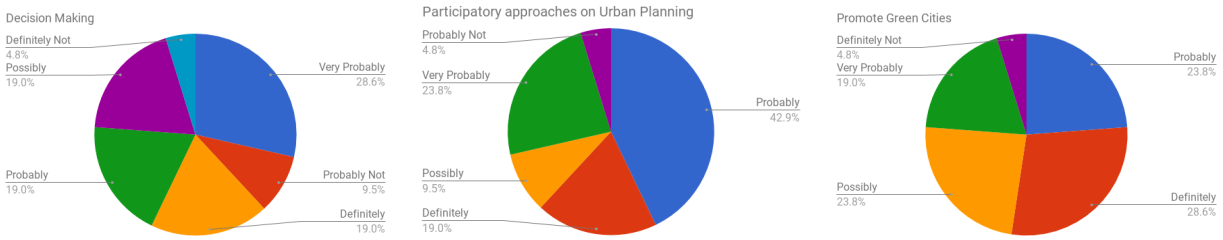
FaceLift is a transparent framework that beautifies existing urban scenes. This translates into two main technical advancements. First, FaceLift is able to generate realistic scenes as opposed to existing approaches based on Generative Adversarial Networks whose transformations are still abstract templates. Second, it augments the deep learning black-box with a module that offers explanations on what has been transformed, making that box more transparent.

There are still important limitations though. One is data bias. The framework is as good as its training data, and more work has to go into collecting reliable ground truth of human perceptions. This data should ideally be stratified according to the people's characteristics that impact their perceptions. The other main limitation is that generative models are hard to control, and more work has to go into offering principled ways of fine-tuning the generative process.

Despite these limitations, FaceLift has the potential to support urban interventions in scalable and replicable ways: it can be applied to the scale of an entire city, and that can be replicated in other cities. The advantage of shifting the focus of research away from predictive analytics towards urban interventions is that people will have technologies that allow them to contribute to discussions on works of architecture more than they can do nowadays. To turn existing spaces into something more beautiful, that will still be the duty of architecture. Yet, with technologies similar to FaceLift more readily integrated in the architecture discussions, the complex job of recreating restorative spaces in an increasingly urbanized world will be greatly simplified. After all, "we delight in complexity to which genius have lent an appearance of simplicity." [3] In the context of future work, that genius is represented by the future technologies that we will contribute to build to deal with the complexity of our cities.

REFERENCES

[1] C. Alexander, S. Ishikawa, and M. Silverstein. 1977. *A Pattern Language: Towns, Buildings, Constructions*. Oxford University Press.
[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. 2015. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv preprint arXiv:1511.00561* (2015).
[3] A. De Botton. 2008. *The Architecture of Happiness*. Knopf Doubleday Publishing Group.
[4] Marco De Nadai, Radu Laurentiu Vieriu, Gloria Zen, Stefan Dragicevic, Nikhil Naik, Michele Caraviello, Cesar Augusto Hidalgo, Nicu Sebe, and Bruno Lepri. 2016. Are Safer Looking Neighborhoods More Lively?: A Multimodal Investigation into Urban Life. In *Proceedings of the ACM on Multimedia Conference (MM)*.
[5] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. 2012. What makes paris look like paris? *ACM Transactions on Graphics* 31, 4 (2012).
[6] Alexey Dosovitskiy and Thomas Brox. 2016. Inverting visual representations with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4829–4837.



(a) **(b)** **(c)**

Figure 10: Urban expert polling on the extent to which an interactive map of "FaceLifted" scenes promotes: (a) decision making; (b) citizen participation in urban planning; and (c) promotion of green cities.

[7] Abhimanyu Dubey, Nikhil Naik, Devi Parikh, Ramesh Raskar, and César A Hidalgo. 2016. Deep learning the city: Quantifying urban perception at a global scale. *arXiv preprint arXiv:1608.01769* (2016).

[8] Reid Ewing and Otto Clemente. 2013. *Measuring urban design: Metrics for livable places*. Island Press.

[9] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. 2017. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028* (2017).

[10] Ralf Herbrich, Tom Minka, and Thore Graepel. 2007. TrueSkillâĎć: a Bayesian skill rating system. In *Advances in neural information processing systems*. 569–576.

[11] J. Jacobs. 1961. *The Death and Life of Great American Cities*. Random House.

[12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[13] Pall Jakob Lindal and Terry Hartig. 2012. Architectural variation, building height, and the restorative quality of urban residential streetscapes. *Journal of Environmental Psychology* (2012).

[14] Nikhil Naik, Scott Duke Kominers, Ramesh Raskar, Edward L Glaeser, and César A Hidalgo. 2017. Computer vision uncovers predictors of physical urban change. *Proceedings of the National Academy of Sciences* 114, 29 (2017), 7571–7576.

[15] Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. 2016. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *Advances in Neural Information Processing Systems*. 3387–3395.

[16] Daniele Quercia, Neil Keith O'Hare, and Henriette Cramer. 2014. Aesthetic capital: what makes London look beautiful, quiet, and happy?. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*. ACM, 945–955.

[17] Alec Radford, Luke Metz, and Soumith Chintala. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434* (2015).

[18] Philip Salesses, Katja Schechtner, and César A Hidalgo. 2013. The collaborative image of the city: mapping the inequality of urban perception. *PloS one* 8, 7 (2013), e68400.

[19] Robert J. Sampson and Stephen W. Raudenbush. 2004. Seeing Disorder: Neighborhood Stigma and the Social Construction of Broken Windows. *Social Psychology Quarterly* 67, 4 (2004).

[20] Chanuki Illushka Seresinhe, Tobias Preis, and Helen Susannah Moat. 2017. Using deep learning to quantify the beauty of outdoor places. *Royal Society open science* 4, 7 (2017), 170170.

[21] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1–9.

[22] Brandon K Vaughn. 2008. Data analysis using regression and multilevel/hierarchical models, by Gelman, A., & Hill, J. *Journal of Educational Measurement* 45, 1 (2008), 94–97.

[23] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. 2014. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*. 487–495.