Universidad Católica San Pablo
Departamento de Ciencia de la Computación
Centro de Investigación e Innovación en Ciencia de la Computación

FGV
FUNDAÇÃO GETULIO VARGAS

Deep Learning Techniques in Urban Security Perception Analysis

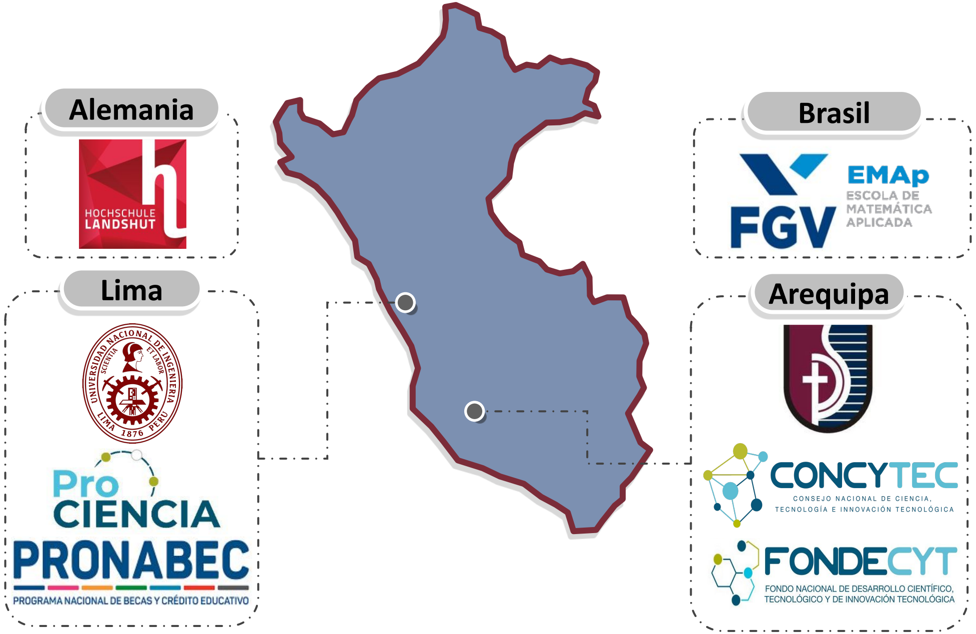
Felipe A. Moreno-Vera
Advisor: Prof. Jorge Poco Medina

FONDECYT
FONDO NACIONAL DE DESARROLLO CIENTÍFICO, TECNOLÓGICO Y DE INNOVACIÓN TECNOLÓGICA
ProCIENCIA
CONCYTEC
CONSEJO NACIONAL DE CIENCIA, TECNOLOGÍA E INNOVACIÓN TECNOLÓGICA

About me

B.Sc. Felipe A. Moreno

www.fmorenovr.com

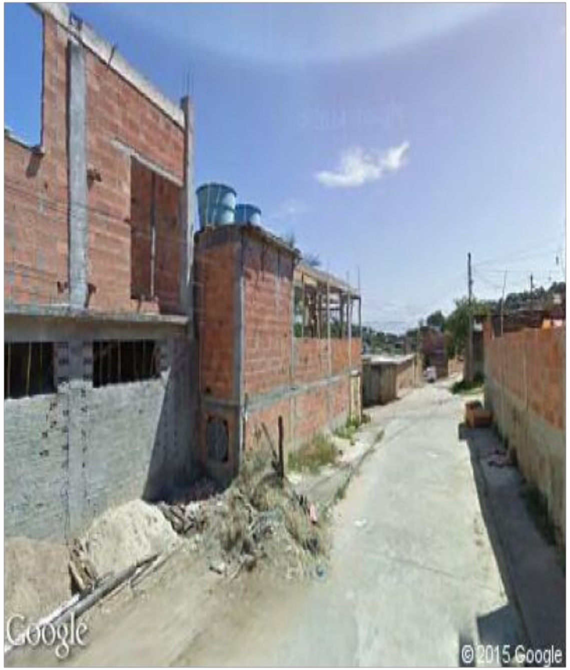


Content

- Context & Motivation
- Dataset
  - Place Pulse
- Methodology
  - Dataset Preparation
  - Exploratory Data Analysis
  - Dataset Limitations
- Urban Safety Perception
  - Data Pre-processing
  - Models Configurations
  - Experiments and Results
- Conclusions

Context & Motivation

Which one looks safer?



Bangú (RJ)

City Center (RJ)

Context

Urban perception is shaped by a complex interplay of factors. Such as physical design, architectural styles, street layouts, landmarks, and the quality of infrastructure all contribute to the visual characteristics that define a city's identity.

Motivation

By understanding how people perceive and experience cities, we can create more inclusive, attractive, and functional urban solutions that meet the needs and aspirations of their diverse populations.

Key contributions

- We **propose a methodology** to analyze the **Place Pulse 2.0 dataset** since we thought that is better to focus on data first instead of model complexity.

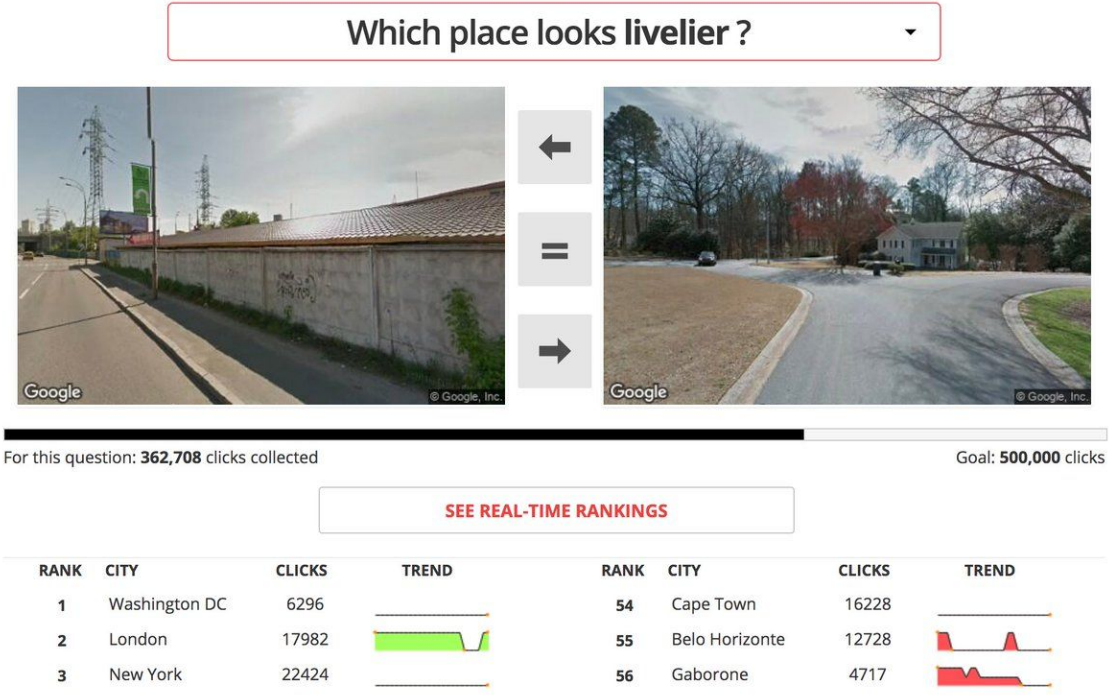
- We **study** the **Place Pulse dataset limitations**, some of them based on how the dataset was built and others based on the pre-processing.

- We **evaluate and compare** semi-supervised models performance with supervised methods, showing that semi-supervised models fit the necessity of training this complex dataset with the limitations explained before.

- We **solved** the problem of imbalance, individual city identification, and lack of samples per city using a semi-supervised GAN model. In other words, we can fix 3 dataset limitations in Place Pulse.

Place Pulse

Place Pulse dataset

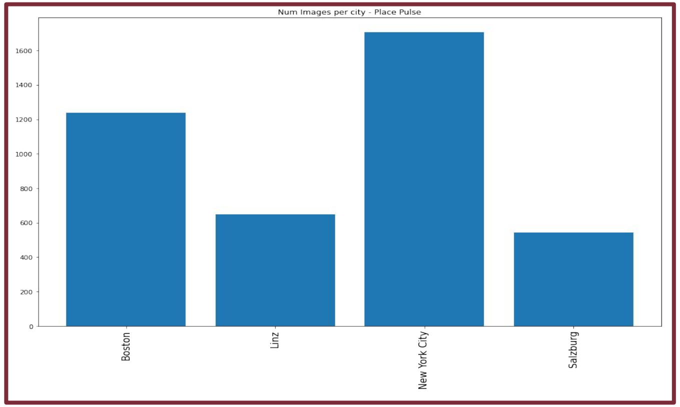


http://pulse.media.mit.edu/

* Comparisons were made using two random images from random cities.

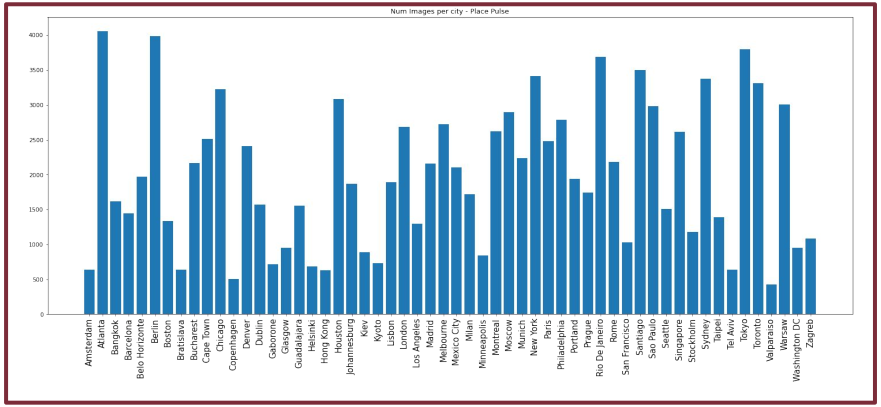
Place Pulse 1.0

- Release date: 2013
- 73 806 Comparisons
- 4 136 images
- 2 Countries
- 4 cities
- 3 categories



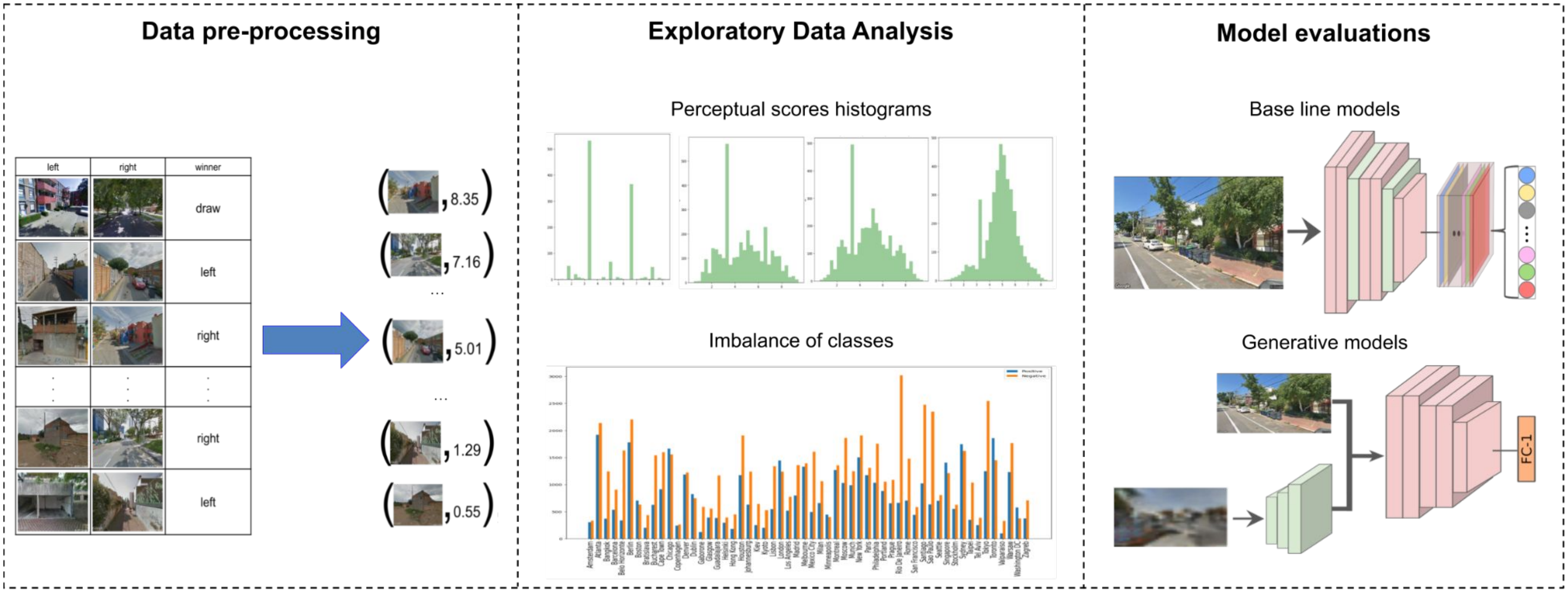
Place Pulse 2.0

- Release date: 2016
- 1 223 649 Comparisons
- 111 390 images
- 32 countries
- 56 cities
- 6 categories



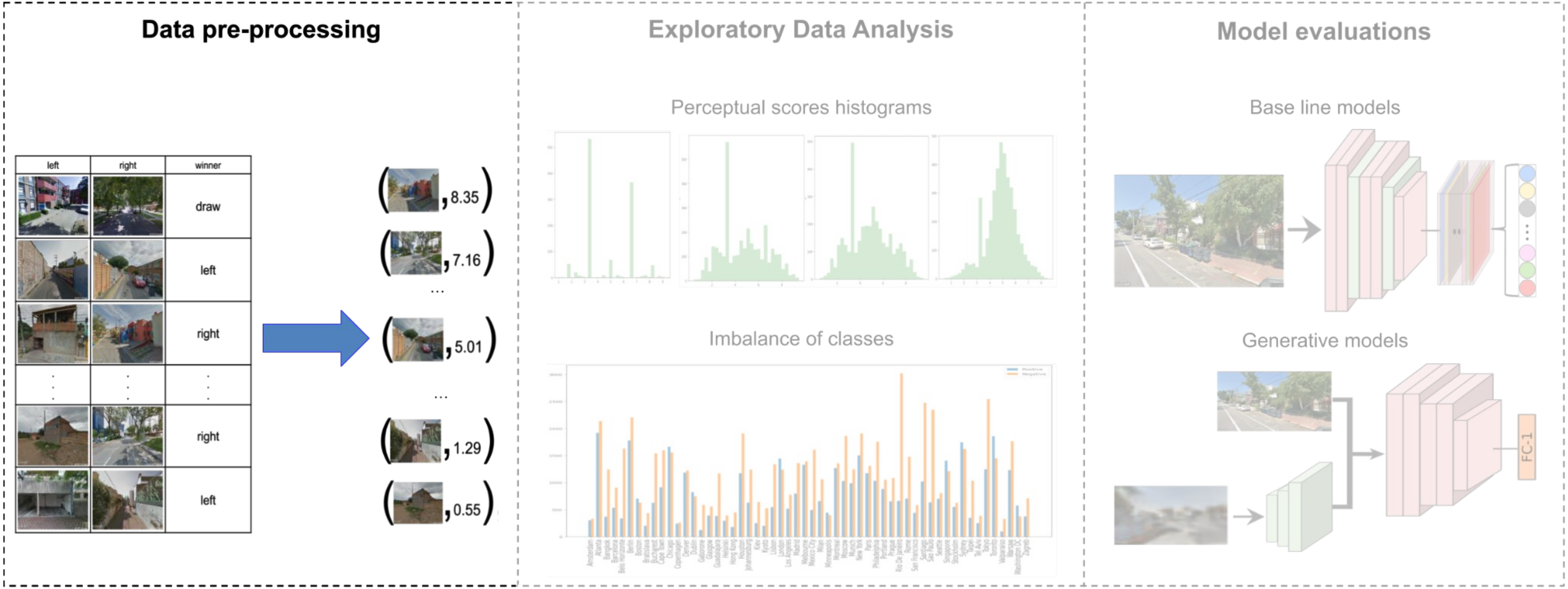
Methodology

Pipeline

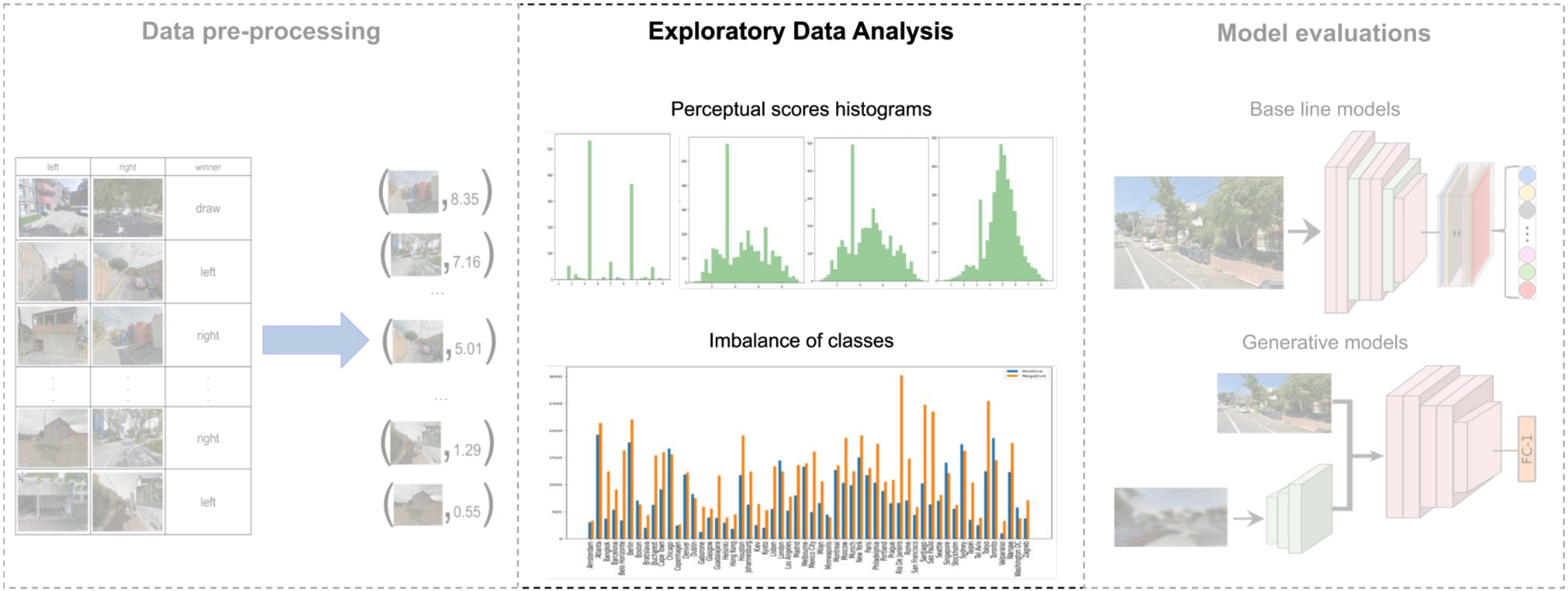


Data pre-processing
left
right
winner
draw
left
right
right
left
8.35
7.16
5.01
1.29
0.55
Exploratory Data Analysis
Perceptual scores histograms
Imbalance of classes
Model evaluations
Base line models
Generative models
FC-1

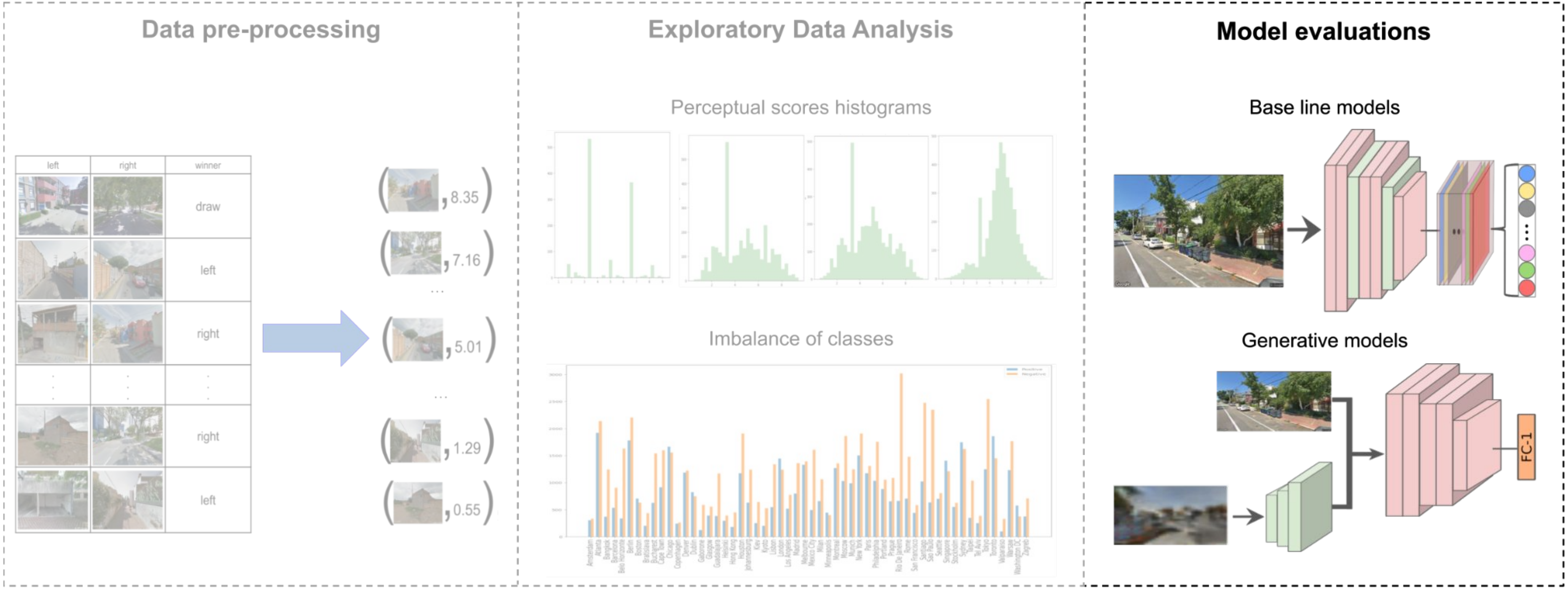
Pipeline


Data pre-processing
left
right
winner
draw
left
right
right
left
, 8.35
, 7.16
, 5.01
, 1.29
, 0.55
Exploratory Data Analysis
Perceptual scores histograms
Imbalance of classes
Model evaluations
Base line models
Generative models
FC-1

Pipeline


Data pre-processing
left
right
winner
draw
left
right
right
left
8.35
7.16
5.01
1.29
0.55
Exploratory Data Analysis
Perceptual scores histograms
Imbalance of classes
Model evaluations
Base line models
Generative models
FC-1

Pipeline



Data pre-processing
left
right
winner
draw
left
right
right
left
8.35
7.16
5.01
1.29
0.55
Exploratory Data Analysis
Perceptual scores histograms
Imbalance of classes
Model evaluations
Base line models
Generative models
FC-1

Data Preparation

Data samples

left-id	right-id	winner	left-lat	left-long	right-lat	right-long	category
513d7e23fdc9f	513d7ac3fdc9f	equal	40.744156	-73.93557	-33.52638	-70.591309	depressing
513f320cfdc9f	513cc3acfdc9f	left	52.551685	13.416548	29.76381	-95.394621	safety
513e5dc3fdc9f	5140d960fdc9f	right	48.878382	2.403116	53.32932	-6.231007	lively

Perceptual Scores

$$W_i = \frac{w_i}{w_i + d_i + l_i}$$

$$L_i = \frac{l_i}{w_i + d_i + l_i}$$

$$q_{i,k} = \overset{*}{\frac{10}{3}}(W_{i,k} + \frac{1}{n^w_{i,k}}(\sum_{j_1} W_{j_1,k}) - \frac{1}{n^l_{i,k}}(\sum_{j_2} L_{j_2,k}) + 1)$$

*Nassar et al, “The evaluative image of the city”, 1990

Salesse et. al, “The Collaborative Image of The City: Mapping the Inequality of Urban Perception”, 2013

Rank Scores

$$\mu_x \longleftarrow \mu_x + \frac{\sigma_x^2}{c} \cdot f\left(\frac{(\mu_x - \mu_y)}{c}, \frac{\varepsilon}{c}\right)$$

$$\mu_y \longleftarrow \mu_y - \frac{\sigma_y^2}{c} \cdot f\left(\frac{(\mu_x - \mu_y)}{c}, \frac{\varepsilon}{c}\right)$$

$$\sigma_x^2 \longleftarrow \sigma_x^2 \cdot \left[1 - \frac{\sigma_x^2}{c} \cdot g\left(\frac{(\mu_x - \mu_y)}{c}, \frac{\varepsilon}{c}\right)\right]$$

$$\sigma_y^2 \longleftarrow \sigma_y^2 \cdot \left[1 - \frac{\sigma_y^2}{c} \cdot g\left(\frac{(\mu_x - \mu_y)}{c}, \frac{\varepsilon}{c}\right)\right]$$

$$c^2 = 2\beta^2 + \sigma_x^2 + \sigma_y^2$$

$$q_{i,k} = \frac{\overset{**}{10}}{c_{max,k}}(c_{i,k})$$

**Minka et al, “TrueSkill 2: An improved Bayesian skill rating system”, 2018

Dubey et. al, “Deep Learning the City : Quantifying Urban Perception At A Global Scale”, 2016

Processed samples

Image	ID	Safety	Lively	Wealthy	Beauty	Boring	Depressive
	513d7e23fdc9f	7.42	8.58	6.5	7.3	2.64	1.23
	513f320cfdc9f	6.07	4.97	7.13	8.61	1.67	0.86

Statistics

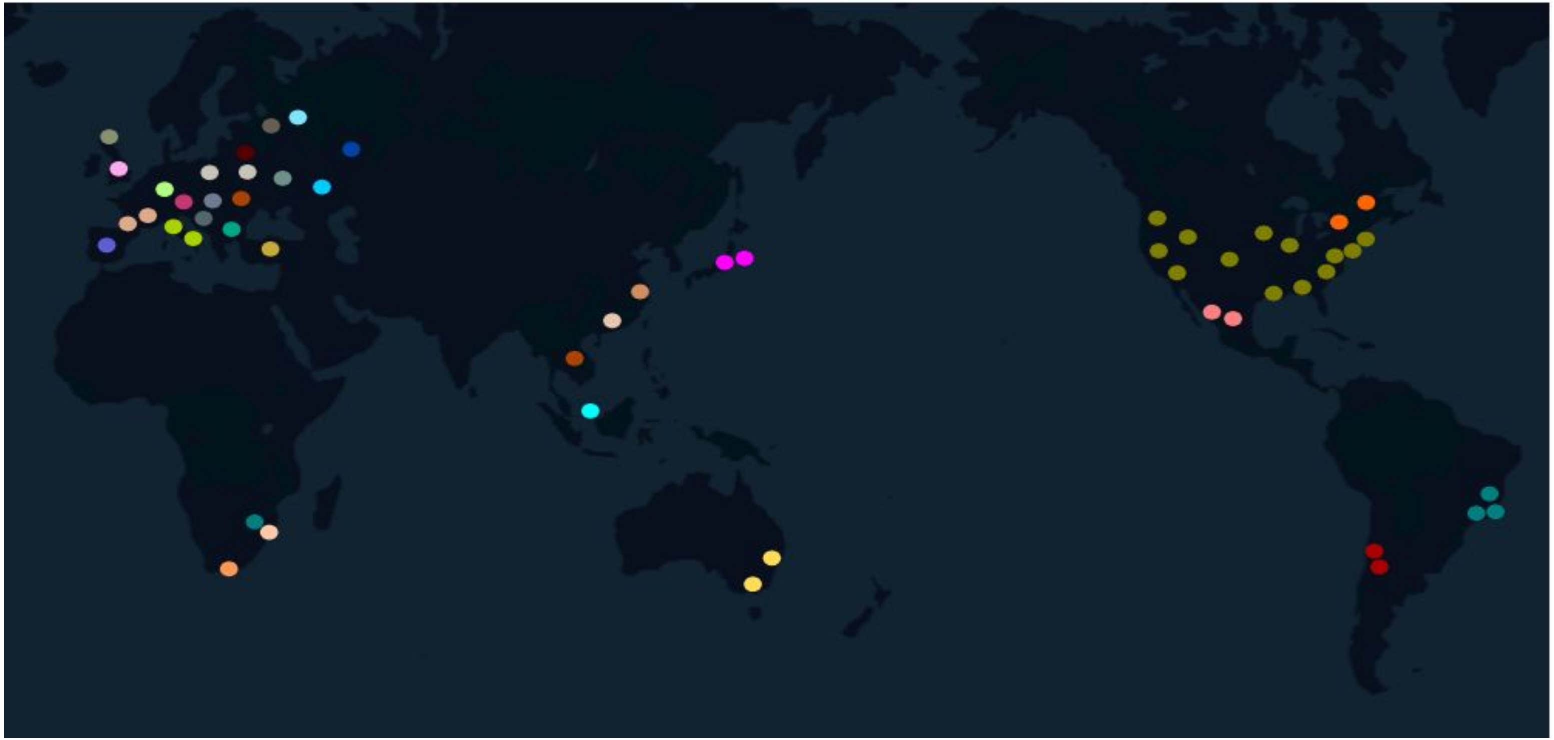
Place Pulse 1.0				
City	# images	*safe mean*	*wealth mean*	*unique mean*
Linz	650	4.85	5.01	4.83
Boston	1237	4.93	4.97	4.76
New York	1705	4.47	4.31	4.46
Salzburg	544	4.75	4.89	5.04
Total	4136			

Place Pulse 2.0			
Continent	#countries	#cities	#images
Europe	19	22	38,747
North America	3	17	37504
South America	2	5	12,524
Asia	5	7	11,417
Oceania	1	2	6,097
Africa	2	3	5,101
Total	32	56	111,390

Place Pulse 2.0			
Category	# comparisons	# images	*mean*
Safety	368,926	111,389	5.188
Lively	267,292	111,348	5.085
Beautiful	175,361	110,766	4.920
Wealthy	152,241	107,795	4.890
Depressing	132,467	105,495	4.816
Boring	127,362	106,363	4.810
Total	1,223,649		

Exploratory Data Analysis

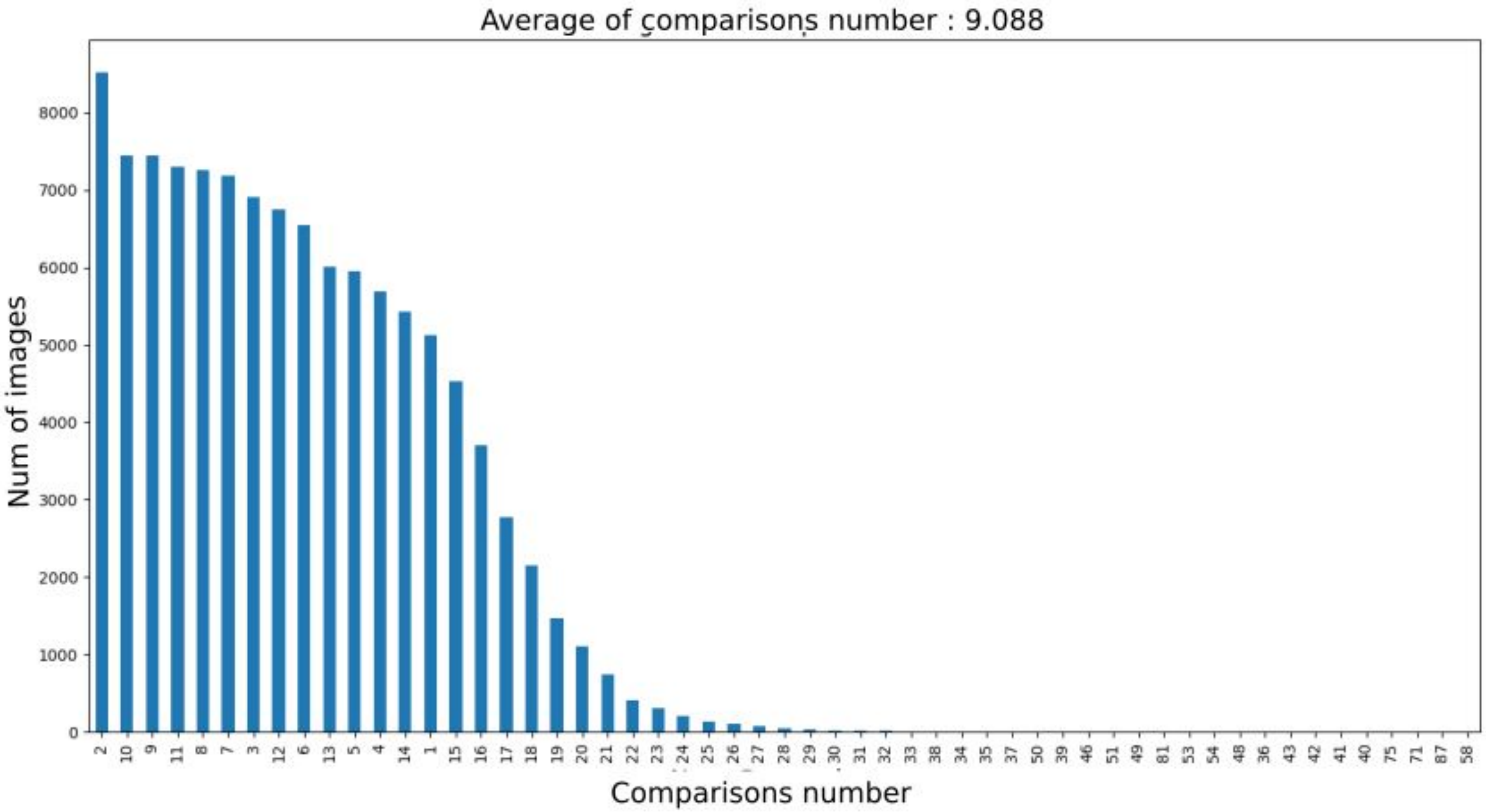
Geographical city distribution



Note: Same color means same country.

Number of images per continent

Number of comparison frequency



High scores images

Low scores images

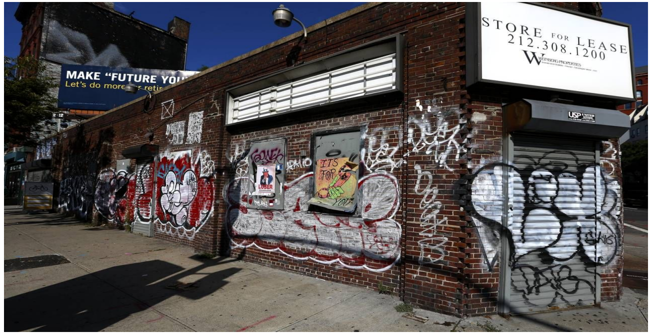
Geographical level vs evaluated images

Place Pulse 2.0				
Category/Level	City	Country	Continent	Global
safety	20,143	45,640	85,890	111,390
lively	14,803	38,216	79,788	111,349
Beautiful	9,410	28,811	66,792	110,767
Wealthy	7,642	24,326	57,780	107,796
Depressing	6,556	21,171	52,504	105,496
Boring	6,148	20,931	52,031	106,364

Dataset Limitations

Individual perception

Safe perception | Unsafe perception



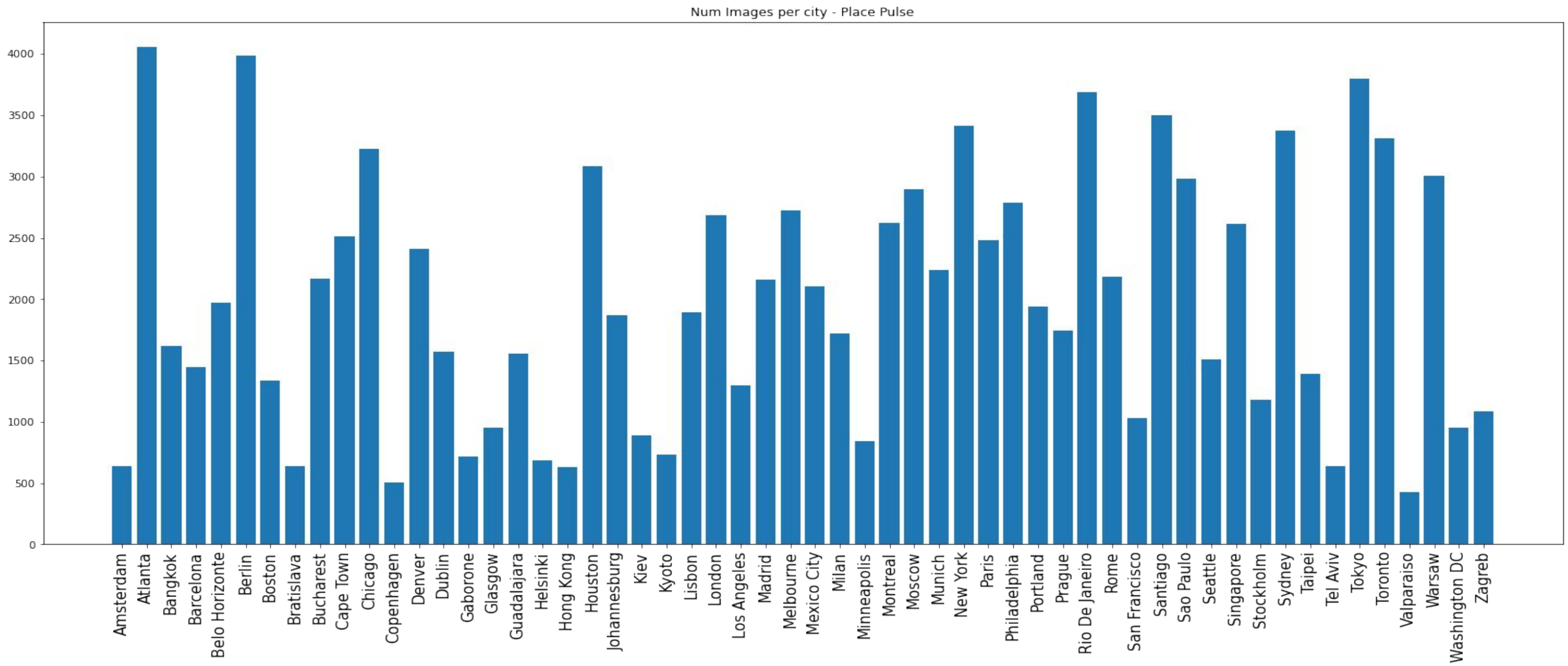
New York*

Tokyo**

*https://www.nytimes.com/2019/08/08/nyregion/newyorktoday/times-square-panic-safety.html#:~:text=Actually%2C%20Times%20Square%20is%20one,23%2C000%20major%20crimes%20were%20recorded.

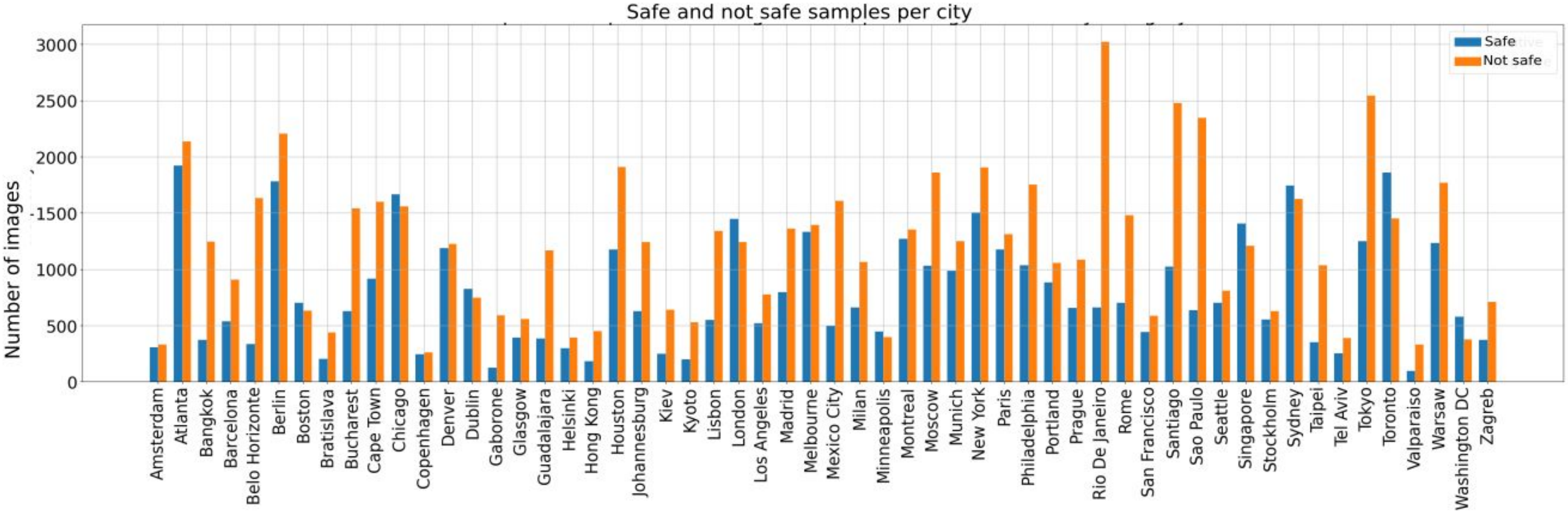
**https://www.japantimes.co.jp/news/2019/10/04/national/media-national/rip-off-bars-japan-tourist-boom/

Lack of samples per city

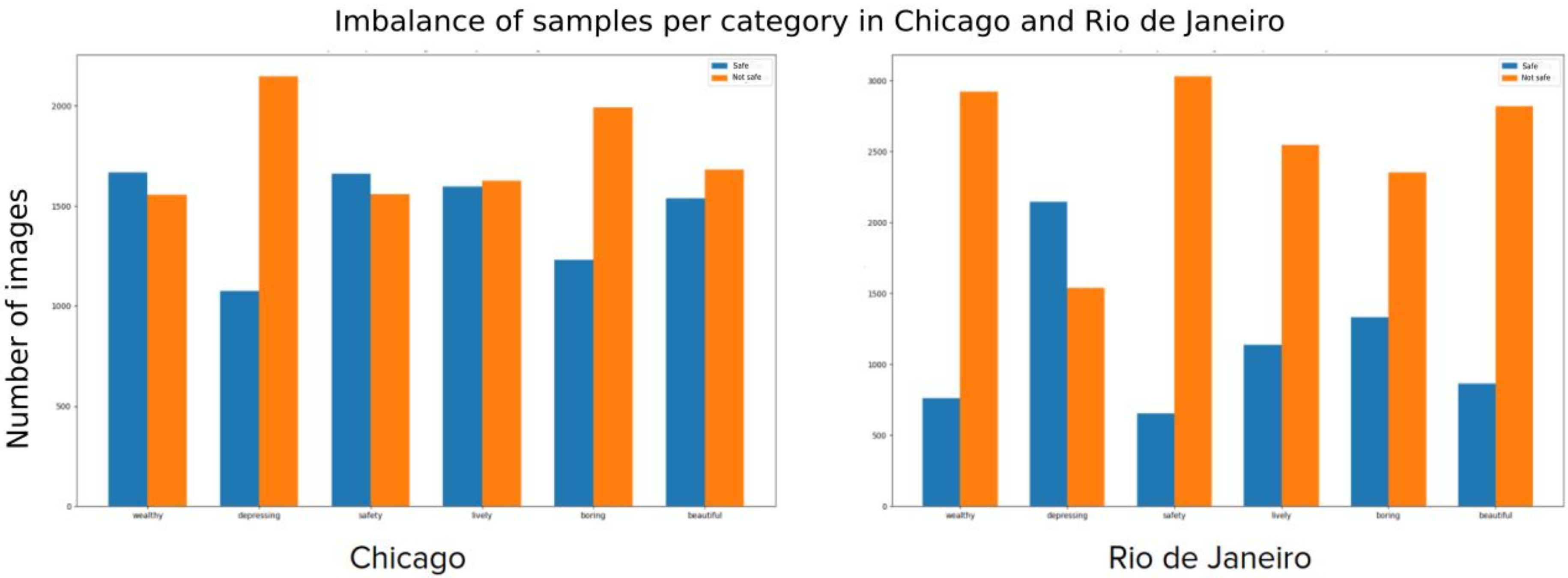


Place Pulse 1.0 < 4 140 Images
Place Pulse 2.0 < 112 000 Images

Imbalance of samples per city



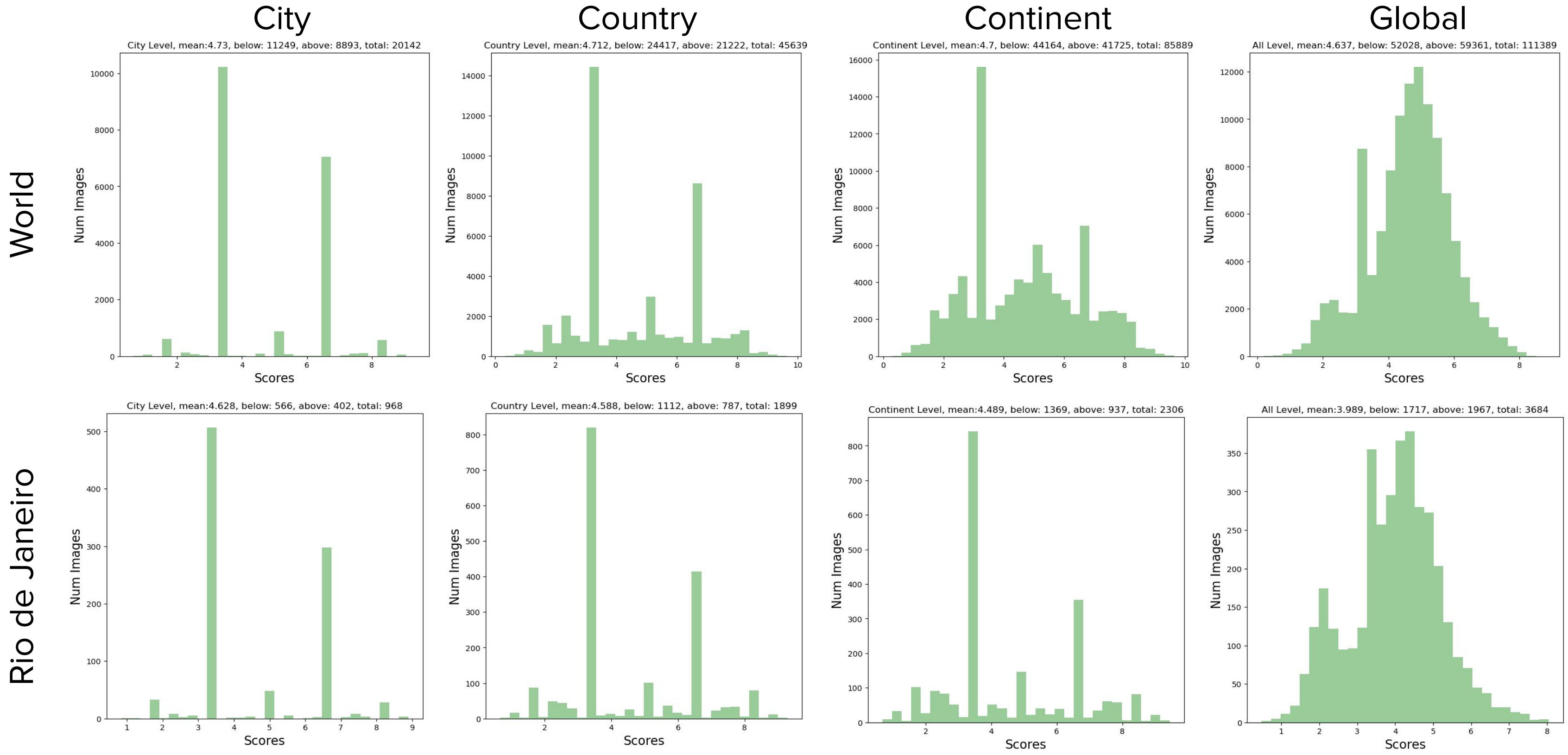
Imbalance of samples



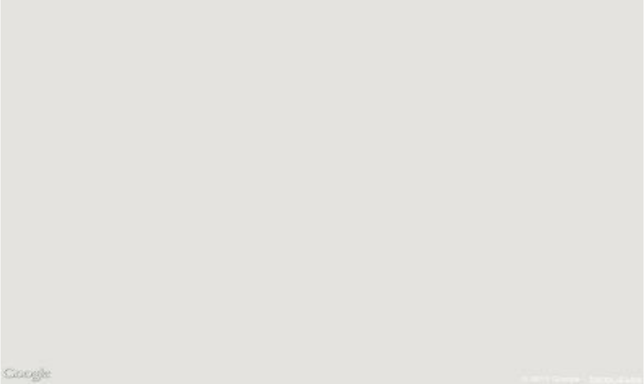
***Positive Samples:** safe, beautiful, wealthy, lively, not depressing, not boring.

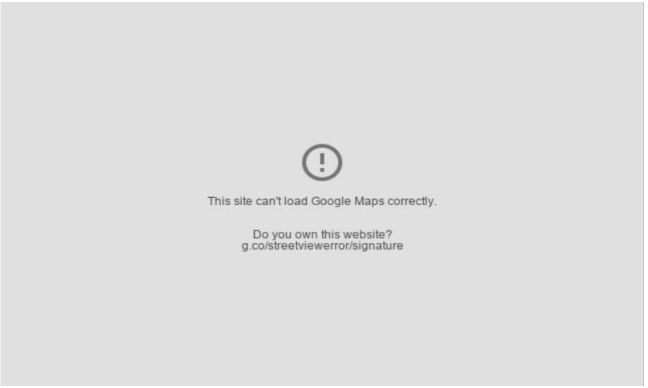
***Negative Samples:** not safe, not beautiful, not wealthy, not lively, depressing, boring.

Non-reliable perceptual scores



Blank/None/Faulty samples

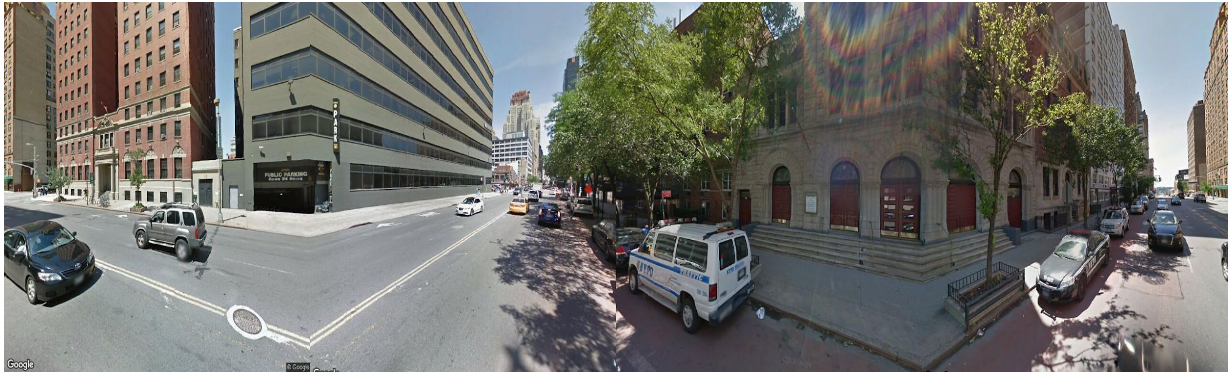




Different point of view (PoV)

Panoramic vs angled samples

Angle: 90



Panoramic

Samples changes over time

ID: 3936

ID: 1

2011 2013 2019

Data Pre-processing

Street View Imagery

Perceptual scores

left	right	winner
		draw
		left
		right
⋮	⋮	⋮
		right
		left

$$\hat{y}_{i,k} = q_{i,k}$$

I: (X,Y)

Data labeling

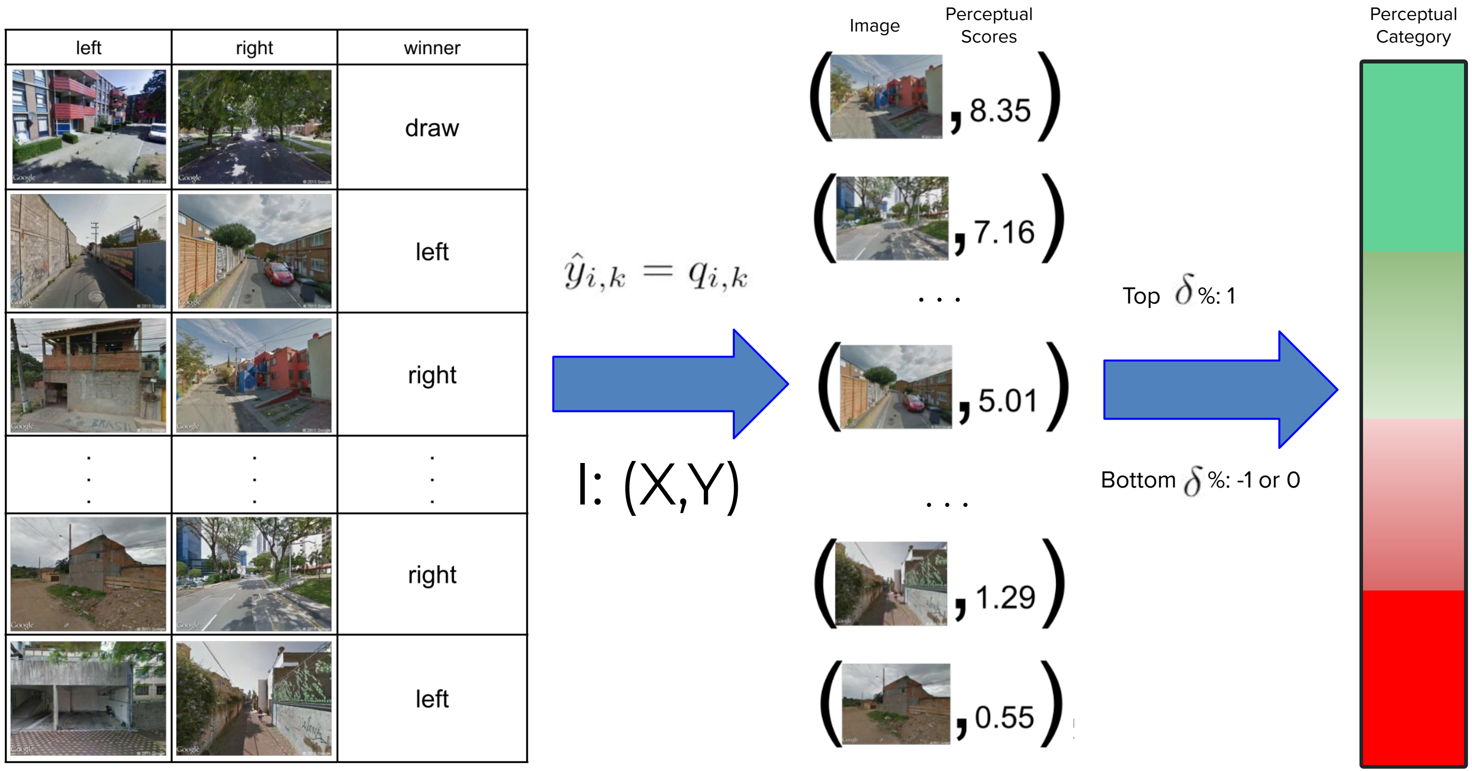
We define a parameter δ which will helps to labeling our data.

$$\hat{y}_{i,k} = q_{i,k}$$

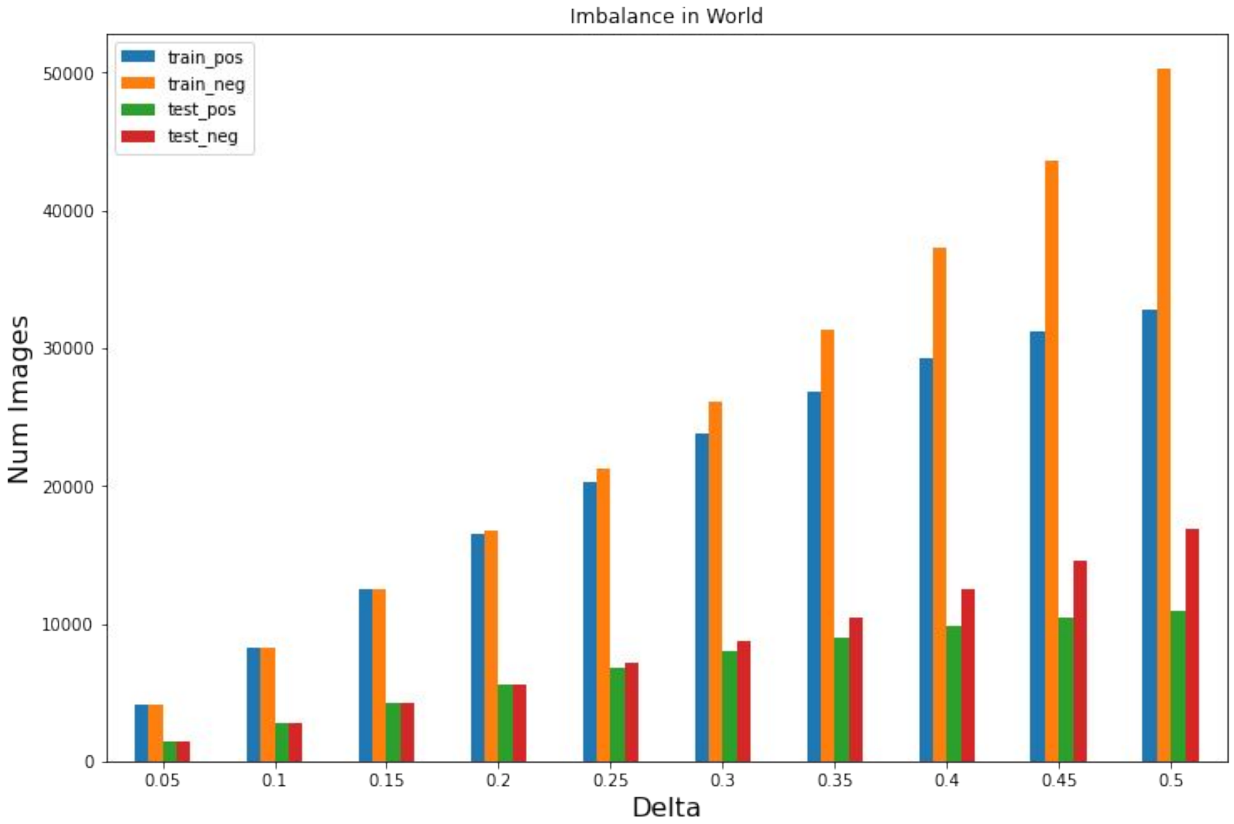
$$\Downarrow$$

$$y_{i,k} = \begin{cases} 1 & \text{if } (q_{i,k}) \text{in the top } \delta\% \\ \text{-}1 & \text{if } (q_{i,k}) \text{in the bottom } \delta\% \end{cases}$$

Perceptual label

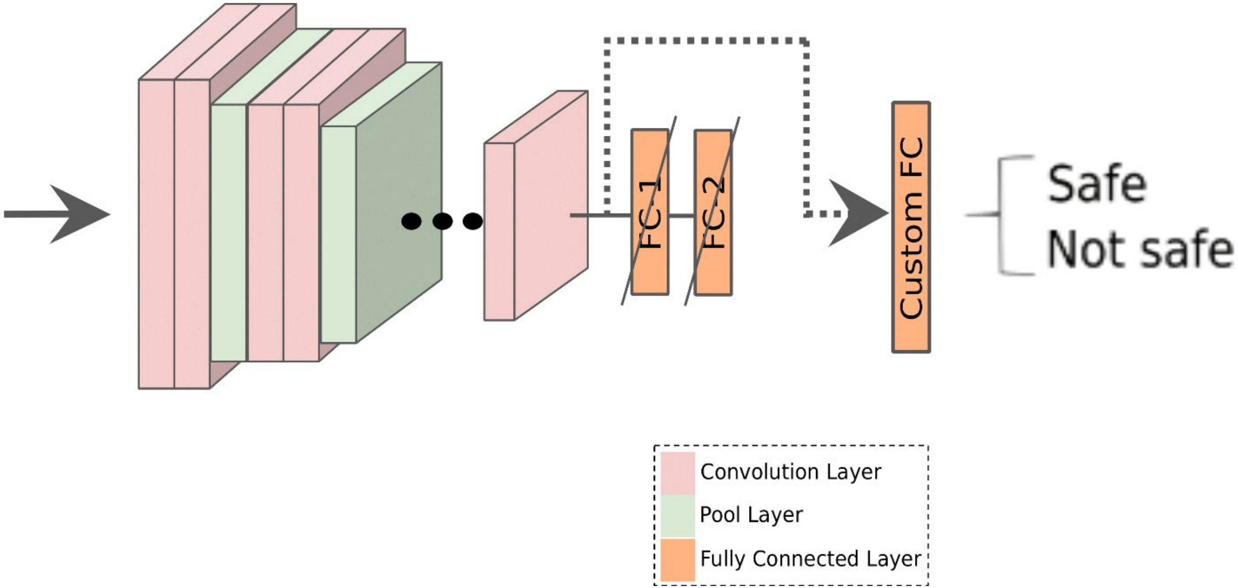


Evaluating δ values



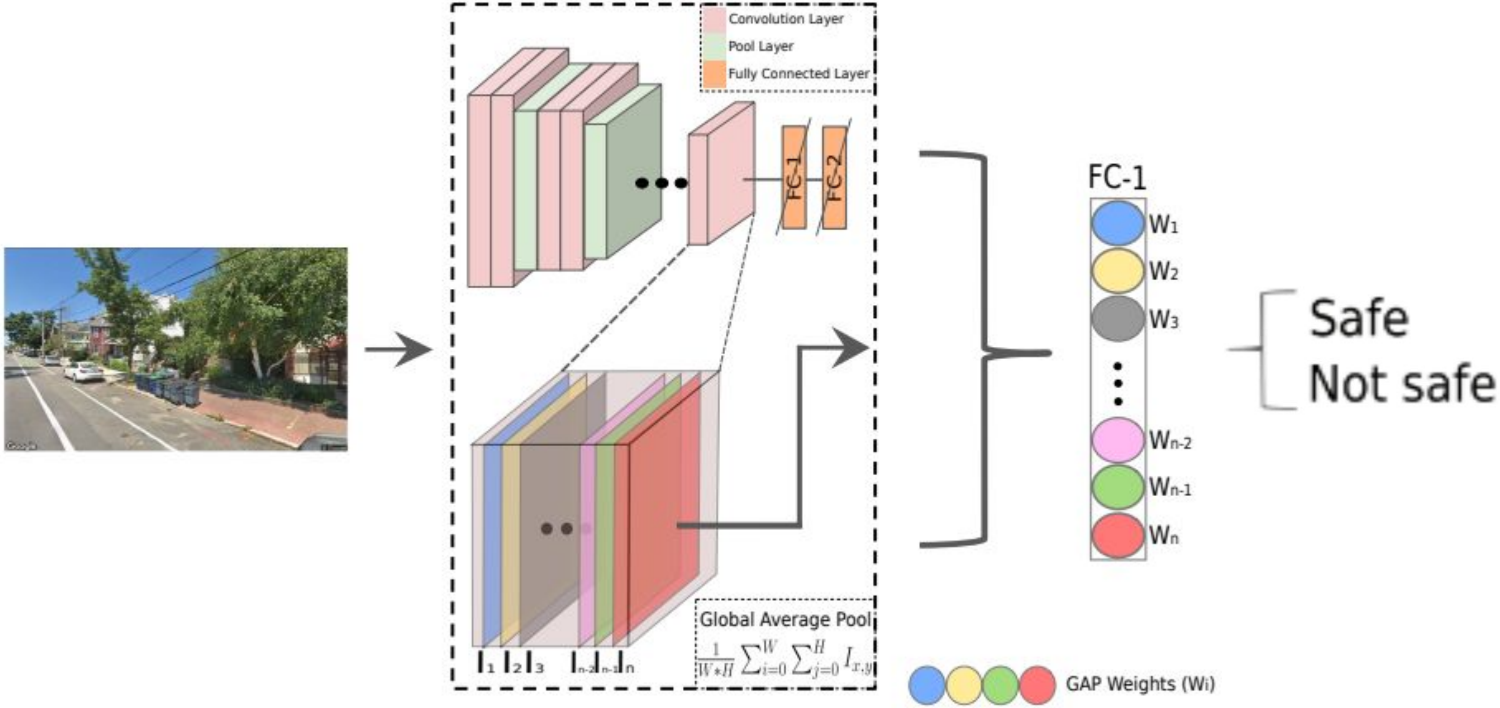
Models Configurations

Transfer learning & Fine-tuning



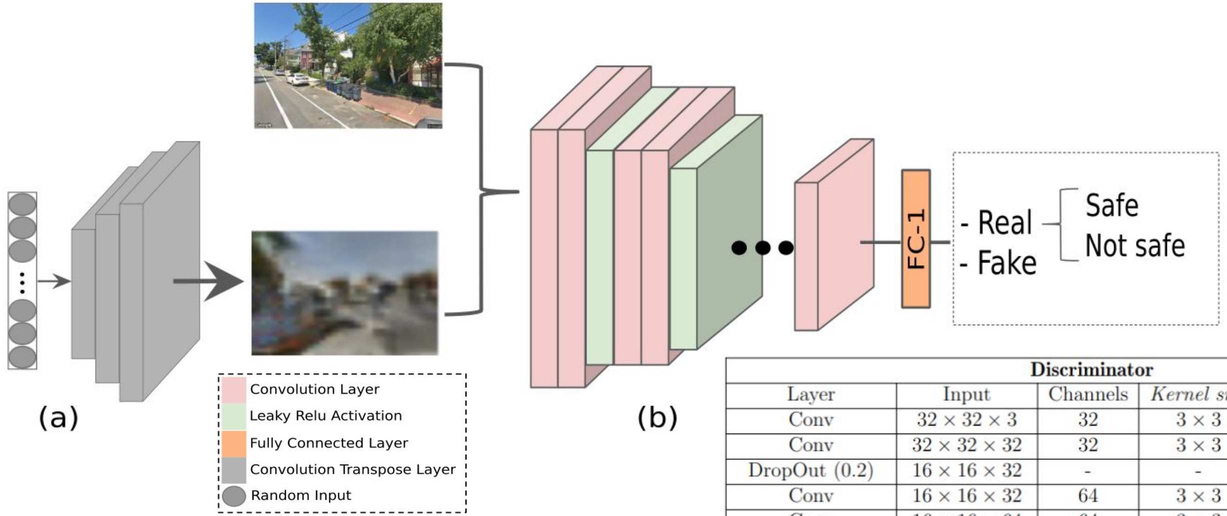
- VGG, ResNet, and Xception
- Input shape: 224x244

Global Average Pooling



- VGG, ResNet, and Xception
- Input shape: 224x244

Generative Adversarial Network (GAN)



Generator					
Layer	Input	Channels	*Kernel size*	*Stride*	Activation
Latent	100	-	-	-	-
Dense	4096	-	-	-	LeakyReLU
Re-shape	$4 \times 4 \times 256$	-	-	-	-
Deconv	$4 \times 4 \times 256$	256	4×4	2	LeakyReLU
Deconv	$8 \times 8 \times 256$	128	4×4	2	LeakyReLU
Deconv	$16 \times 16 \times 128$	64	4×4	2	LeakyReLU
Conv	$32 \times 32 \times 64$	3	3×3	1	Tanh
Total parameters	2 119 811				

Discriminator					
Layer	Input	Channels	*Kernel size*	*Stride*	Activation
Conv	$32 \times 32 \times 3$	32	3×3	1	LeakyReLU
Conv	$32 \times 32 \times 32$	32	3×3	2	LeakyReLU
DropOut (0.2)	$16 \times 16 \times 32$	-	-	-	-
Conv	$16 \times 16 \times 32$	64	3×3	1	LeakyReLU
Conv	$16 \times 16 \times 64$	64	3×3	2	LeakyReLU
DropOut (0.2)	$8 \times 8 \times 64$	-	-	-	-
Conv	$8 \times 8 \times 64$	128	3×3	1	LeakyReLU
Conv	$8 \times 8 \times 128$	128	3×3	2	LeakyReLU
DropOut (0.2)	$4 \times 4 \times 128$	-	-	-	-
Conv	$4 \times 4 \times 128$	256	3×3	1	LeakyReLU
Flatten	$4 \times 4 \times 256$	-	-	-	-
Dense	128	-	-	-	-
DropOut (0.4)	128	-	-	-	-
Dense	3	-	-	-	Softmax
Total parameters	1 107 882				

Model hyperparameters

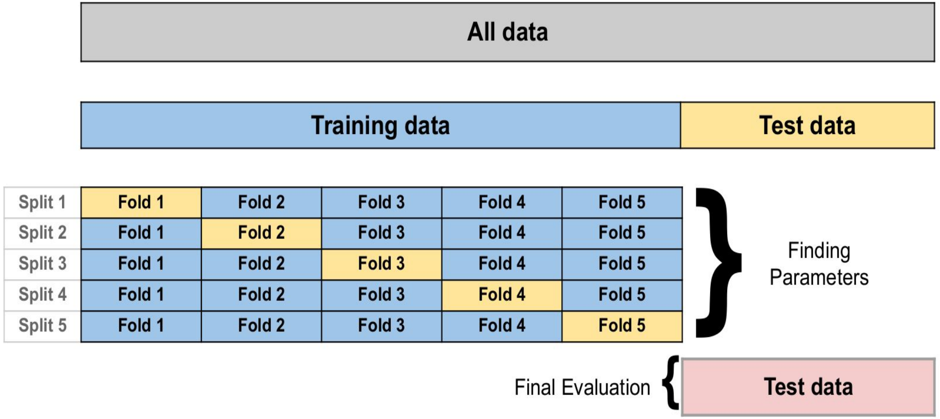
Summary of model parameters							
Name	**Model hyperparameters**						**Data**
Method	Input	*Batch*	*Opt*	*LR*	*Ep/It*	*CV*	Geo. level
TL_VGG	4096	-	lbfgs	-	1000	5	Global/City
TL_VGG_GAP	512	-	lbfgs	-	1000	5	Global//city
FT_VGG	$224 \times 224 \times 3$	128	Adam	$1e^{-3}$	100	5	Global/City
FT_VGG_GAP	$224 \times 224 \times 3$	128	Adam	$1e^{-3}$	100	5	Global/City
SSL_GAN_Dis	$32 \times 32 \times 3$	128	Adam	$1e^{-3}$	100	5	Global
SSL_GAN_Gen	100	128	Adam	$1e^{-3}$	100	5	Global

* Parameters were found using GridSearchCV.
* Trained on GPU NVIDIA GeForce GTX 1070, 8 Gb VRAM.
* EarlyStop in 30 epochs and DecayLR every 8 epochs.

Data split

- Oversampling method to balance classes and split data into 75% and 25%, respectively

- Hyperparameters tuning: Grid search using Stratified 5 Cross-Validation



Metrics

- Accuracy — What percent of the data were predicted correct?
- Precision — What percent of your predictions were correct?
- Recall — What percent of the positive cases did you catch?
- F1 score — What percent of positive predictions were correct?

$$Accuracy = \frac{T_P + T_N}{T_P + T_N + F_P + F_N}$$

$$Recall = \frac{T_P}{T_P + F_N} \qquad Precision = \frac{T_P}{T_P + F_P}$$

$$F1_{score} = 2\frac{Precision * Recall}{Precision + Recall}$$

Experiments and Results

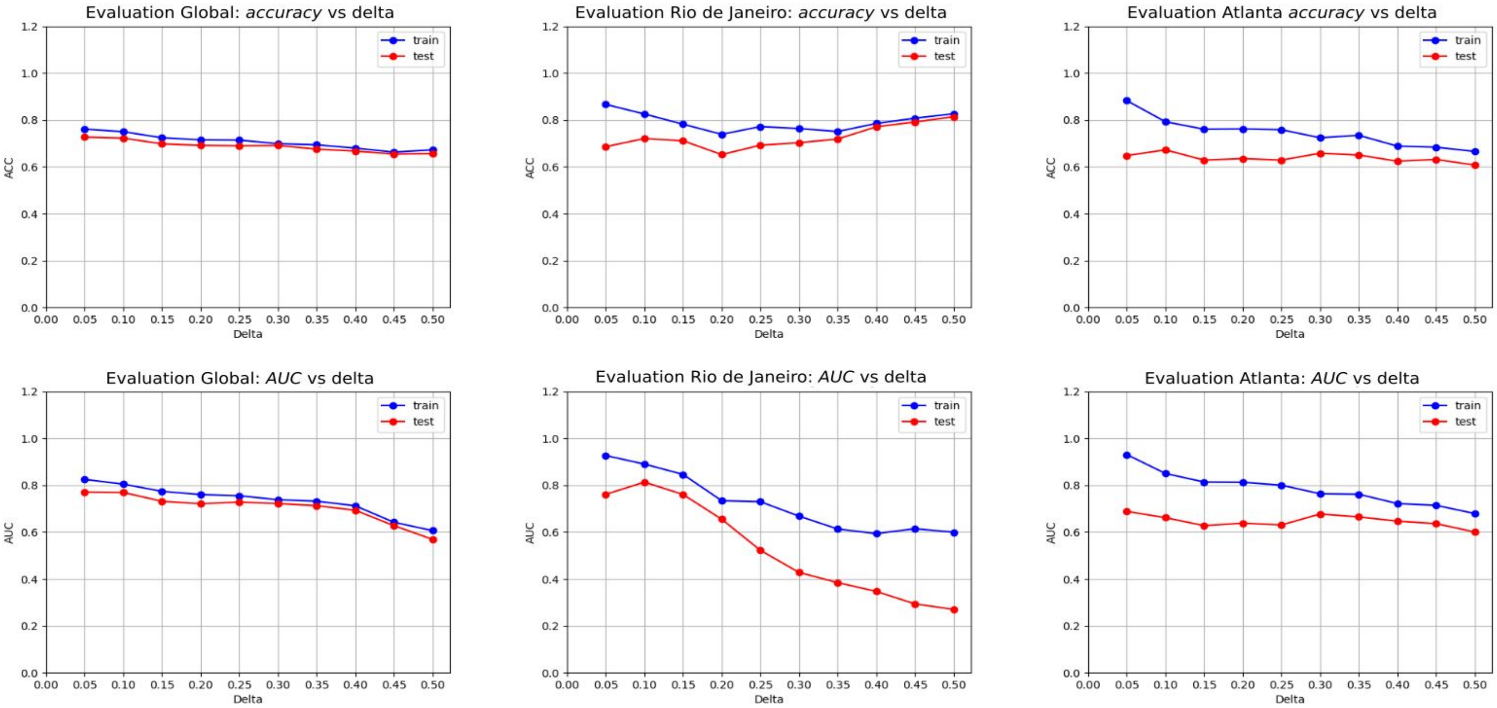
Publications

- **Quantifying Urban Safety Perception on Street View Images**
  Felipe Moreno-Vera, Bahram Lavi, and Jorge Poco. *In IEEE/WIC/ACM International Conference on Web Intelligence (WI-IAT)*, 2021.

- **Urban Perception: Can We Understand Why a Street Is Safe?**
  Felipe Moreno-Vera, Bahram Lavi, and Jorge Poco. *In Mexican International Conference on Artificial Intelligence (MICAI)*, 2021.

- **Understanding Safety based on Urban Perception**
  Felipe Moreno-Vera. *In International Conference on Intelligent Computing (ICIC)*, 2021.

Transfer learning



* Results of testing using different values of δ.

Transfer learning

		auc		*accuracy*		*f1 score*	
Model	Method	train	eval	train	eval	entrena	eval
VGG	*LinearSVC*	63.62	56.50	68.85	65.22	54.78	**49.41**
	Logistic	60.63	**57.52**	67.25	**65.72**	51.42	49.07
	Ridge Classifier	64.72	54.75	69.44	64.38	56.50	49.34
	RBF SVC	45.14	42.42	52.13	52.37	46.93	46.59
VGG_GAP	*LinearSVC*	59.01	**57.93**	66.51	**66.09**	49.52	49.06
	Logistic	58.07	57.57	65.95	65.59	46.06	45.61
	Ridge Classifier	59.20	57.93	66.59	65.89	50.27	**49.76**
	RBF SVC	42.93	41.70	50.25	50.35	47.16	46.75
VGG_Places	*LinearSVC*	64.44	57.14	69.48	65.79	56.39	51.20
	Logistic	61.74	**58.35**	68.16	**66.44**	53.77	**51.28**
	Ridge Classifier	65.20	55.76	69.84	64.86	57.56	50.67
	RBF SVC	47.32	45.25	56.56	55.69	44.78	44.21

Transfer learning

Model	Method	*auc* train	*auc* eval	*accuracy* train	*accuracy* eval	*f1 score* entrena	*f1 score* eval
VGG_GAP_Places	*LinearSVC*	60.26	**59.76**	67.38	**66.96**	51.65	51.04
	Logistic	59.40	58.97	66.81	66.62	49.16	48.90
	Ridge Classifier	60.45	59.15	67.45	66.94	52.23	**51.53**
	RBF SVC	44.40	42.47	52.59	52.54	43.39	45.05
ResNet50	*Linear SVC*	61.62	59.10	68.10	**66.42**	53.63	50.80
	Logistic	60.04	**59.15**	67.25	66.37	51.47	49.70
	Ridge Classifier	62.11	58.38	68.36	66.08	54.59	**51.00**
	RBF SVC	45.36	44.07	53.46	53.57	44.99	44.98
Xception	*LinearSVC*	55.29	**53.25**	64.43	**63.33**	41.66	39.69
	Logistic Regression	53.48	52.75	63.56	63.14	36.72	35.87
	Ridge Classifier	57.23	52.22	65.22	63.04	45.63	42.11
	RBF SVC	45.575	44.99	49.12	49.12	55.01	**55.05**

Fine-tuning

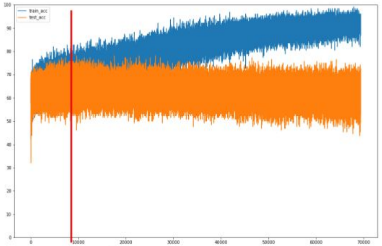
	auc		*accuracy*		*f1 score*	
Models "FT"	train	eval	train	eval	train	eval
VGG	77.83	**77.42**	74.01	64.71	74.01	64.69
VGG_GAP	76.145	75.59	69.40	66.88	69.41	66.87
VGG_Places	77.98	77.35	70.52	**67.28**	70.52	**67.28**
VGG_GAP_Places	74.95	74.75	68.71	67.26	68.71	67.27
ResNet50	76.362	72.71	70.36	65.64	67.35	64.98

GAN

Model 32x32x3	CV	*auc* train	*auc* eval	*accuracy* train	*accuracy* eval	*f1 score* train	*f1 score* eval
SSL_GAN	0	80.95	80.97	90.26	59.06	90.26	59.04
	1	81.43	81.45	89.42	61.50	89.42	61.48
	2	81.43	**81.45**	89.56	**62.58**	89.56	**62.57**
	3	80.59	80.66	90.01	61.52	90.01	61.54
	4	80.61	80.63	89.38	61.14	89.38	61.13

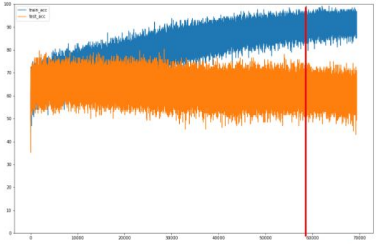
Model 32x32x3	CV	iteration	*auc* train	*auc* eval	*accuracy* train	*accuracy* eval	*f1 score* train	*f1 score* eval
SSL_GAN	0	23788	73.89	73.89	78.90	78.12	78.90	78.12
	1	58550	80.21	**80.22**	92.18	**81.25**	92.18	**81.25**
	2	21951	73.60	73.60	81.25	79.68	81.25	79.68
	3	23180	73.53	73.53	76.56	78.90	76.56	78.90
	4	8602	69.84	69.84	74.21	78.90	74.21	78.90

cv 4



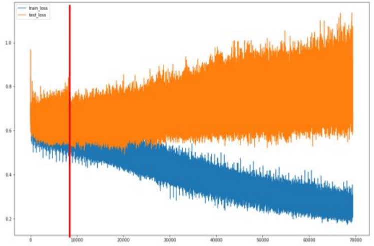
Train-Test Accuracy

cv 1



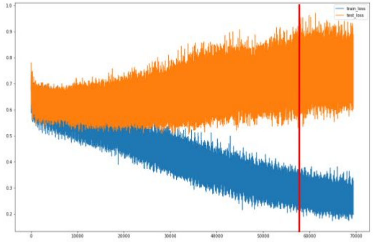
Train-Test Accuracy

cv 4



Train-Test Loss

cv 1



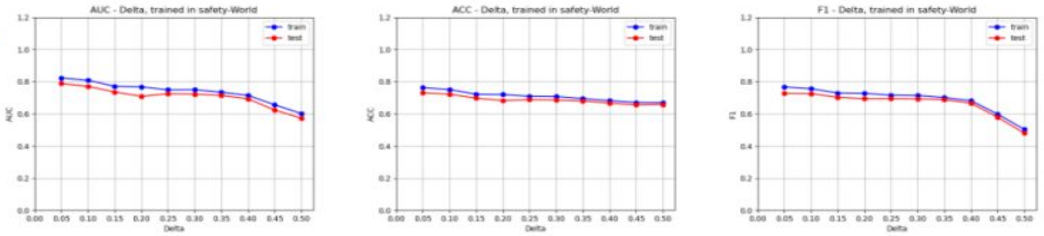
Train-Test Loss

GAN: generated images

Website

Metodo	Metrics		all
RidgeClassifier	Train set	AUC	0.6018
		ACC	0.67356
		F1	0.51896
	Test set	AUC	0.59534
		ACC	0.66687
		F1	0.50497
LogisticRegressi			
LinearSVC			
rbfSVC			

features: vgg16, city: World, metric: safety, and method RidgeClassifier



delta = 0.05	AUC=0.78898	AUC=0.78898	AUC=0.78898	AUC=0.78666	AUC=0.78666	AUC=0.78666	AUC=0.78898	AUC=0.78898	AUC=0.78898	AUC=0.78898	mean(AUC)=0.78828 (0.00106) mean(ACC)=0.73084 (0.00066) mean(F1)=0.72762 (0.0014)
delta = 0.1	AUC=0.76962	AUC=0.77193	AUC=0.76962	AUC=0.77193	AUC=0.76962	AUC=0.77193	AUC=0.76962	AUC=0.77193	AUC=0.76962	AUC=0.76962	mean(AUC)=0.77054 (0.00113) mean(ACC)=0.72252 (0.00132) mean(F1)=0.72578 (0.00119)
delta = 0.15	AUC=0.73549	AUC=0.73549	AUC=0.73587	AUC=0.73587	AUC=0.73587	AUC=0.73587	AUC=0.73549	AUC=0.73587	AUC=0.73549	AUC=0.73549	mean(AUC)=0.73568 (0.00019) mean(ACC)=0.69708 (0.00078) mean(F1)=0.70267 (0.00022)
delta = 0.2	AUC=0.71029	AUC=0.70825	AUC=0.71029	AUC=0.70825	AUC=0.70825	AUC=0.70825	AUC=0.71029	AUC=0.70825	AUC=0.71029	AUC=0.70825	mean(AUC)=0.70907 (0.001) mean(ACC)=0.68263 (0.00123) mean(F1)=0.69411 (0.00057)

Training time

	Training time for each model	
Method	Data	Average Time
SSL_GAN	Global	1 and a half week
FT_VGG	Global	8 hours
FT_VGG	56 Cities	6 hours
FT_VGG_GAP	Global	7 hours
FT_VGG_GAP	56 Cities	5 hours
TL_VGG	Global	15 minutes
TL_VGG	56 Cities	10 minutes
TL_VGG_GAP	Global	9 minutes
TL_VGG_GAP	56 Cities	6 minutes

Conclusions

Conclusions

- We were able to analyze the safety perception using the **Place Pulse dataset**

- It is feasible to **train** a classifier to **infer** the **safe perception** in **street view images.**

- **Black-box CNN models** help to classify images.
  - It performs better than decision trees, SVM, ridge regression, booster models, etc.
  - We won't be able to use complex architectures, such as transformers, due to limited computational resources.

- It has **high performance** in distinguishing safe perceptions between images, it is possible to improve results using generative models such as semi-supervised GAN.

Thanks! Any questions?

felipe.moreno@ucsp.edu.pe

THANKS!