Workshop Topics, Tools and

Model Intros

Workshop Topics and Tools

Overview

task & topic

Develop prototypes of tools and methods to improve urban planning and design practice

tools

Several ML models and workflows for python and Grasshopper

data

3D city model as the playground for your project.

Workshop Topics and Tools

Overview

Day 1 (Sun 28th)

Name	Room	Time (GMT)	Description
Talk: Intro	Webinar	10:00 - 10:15	The workshop structure, introduction to Hackathons, expectations and deliverables
Exercise: Ice Breaker	Webinar	10:15 - 10:45	"A picture a person"
Talk: Intro to CIL	Webinar	10:45 - 11:00	Who we are and what we do
Talk: Intro to AI	Webinar	11:00 - 11:45	Scope, history, heuristics
Break	break	11:45 - 11:55	-
Talk: Workshop Topics and Tools	Webinar	11:55 - 12:40	ML Models, sample problems, playground city model
Break	break	12:40 - 12:45	-
Exercise: Group Forming	Meeting	12:45 - 14:00	Brainstorming, algorithmic groups, idea finding and pitching

CIL CITY INTELLIGENCE LAB

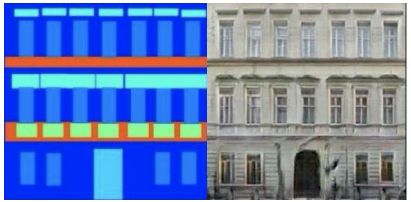
Workshop Topics and Tools

Machine learning models and workflows

Generative Adversarial Networks (GAN)

Pix2pix

Generates new images based on an input image

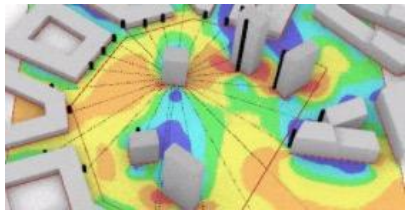


- Approximation of results of traditional optimization engines
- Generation of Buildings based on block outlines
- Rapid visualization of sketches (eg. substitute 3D render engines)

Reinforcement learning (RL)

Deep-Q-learning

Trains a decision making and strategy developing agent

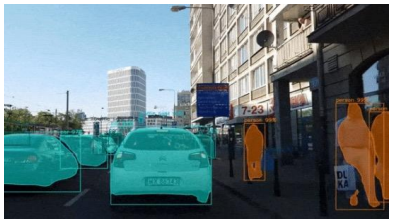


- Simulate crowd behavior with smart RL agents
- Simulate shortcut behavior of of pedestrians
- Manage and stabilize dynamic systems (eg. traffic lights and flows, lockdown policies)

Convolutional Neural Networks (CNN)

Mask R-CNN model

Finds and labels objects in images



- Quantify and understand the content of images
- Predicting the usage of bike sharing schemes and times of load.
- … use the results as training data for other models such as pix2pix or DQL-Agent training

Generative Adversarial Networks (GAN)

Style Transfer

Changes the look of images based on a reference image

- Advance visual quality of images
- Answer questions such as: how would this pace look like in winter, summer or during rain?

grasshopper (train + apply models)

google colab (apply pre-trained models)

Workshop Topics and Tools

pix2pix - what does it do?

Generative Adversarial Networks (GAN)

Pix2pix

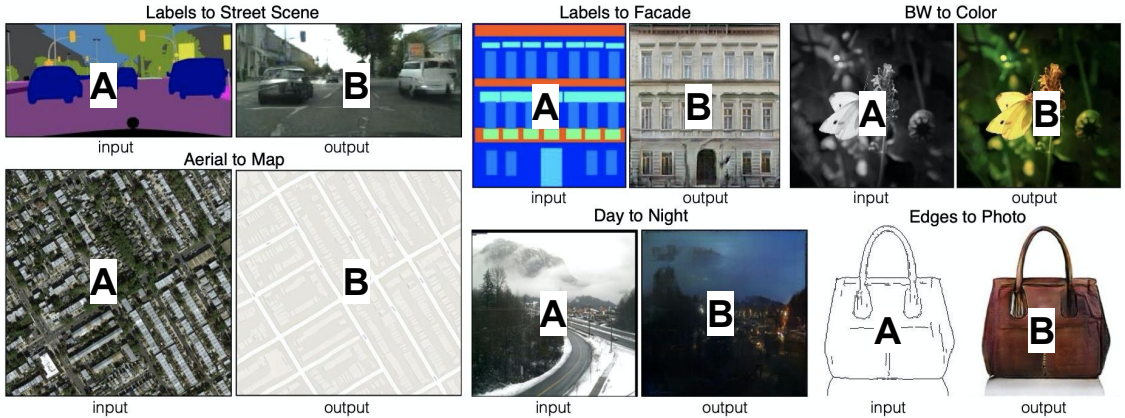
Generates new images based on an input image

Image-to-Image Translation with Conditional Adversarial Networks

Phillip Isola Jun-Yan Zhu Tinghui Zhou Alexei A. Efros

Berkeley AI Research (BAIR) Laboratory, UC Berkeley

{isola,junyanz,tinghuiz,efros}@eecs.berkeley.edu



https://arxiv.org/pdf/1611.07004.pdf

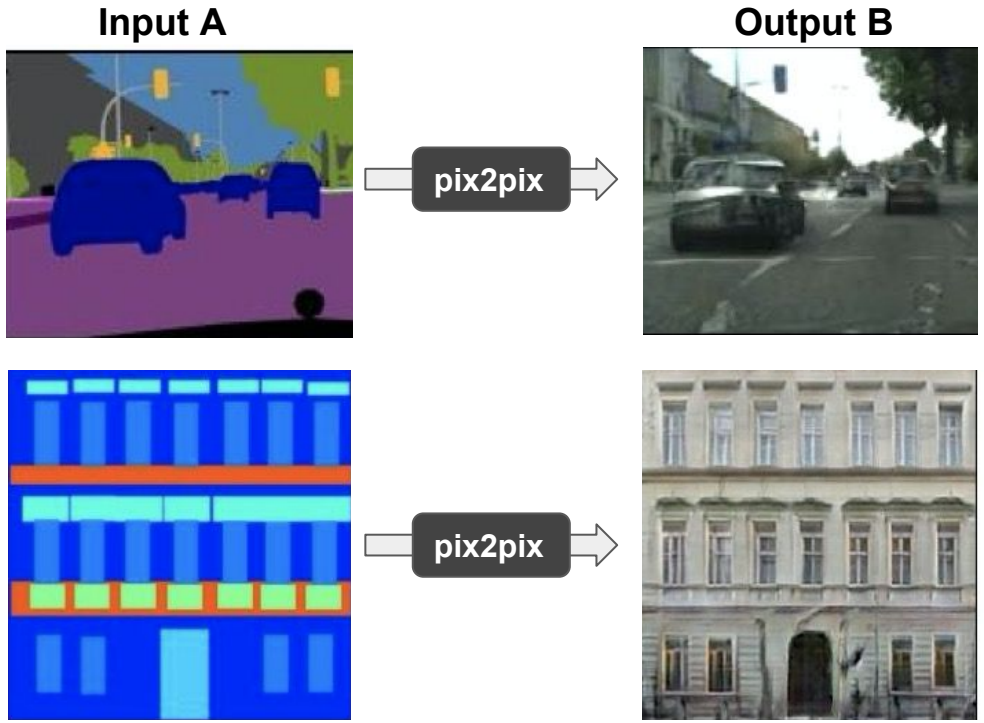
Workshop Topics and Tools

pix2pix - what does it do?

Generative Adversarial Networks (GAN)

Pix2pix

Generates new images based on an input image



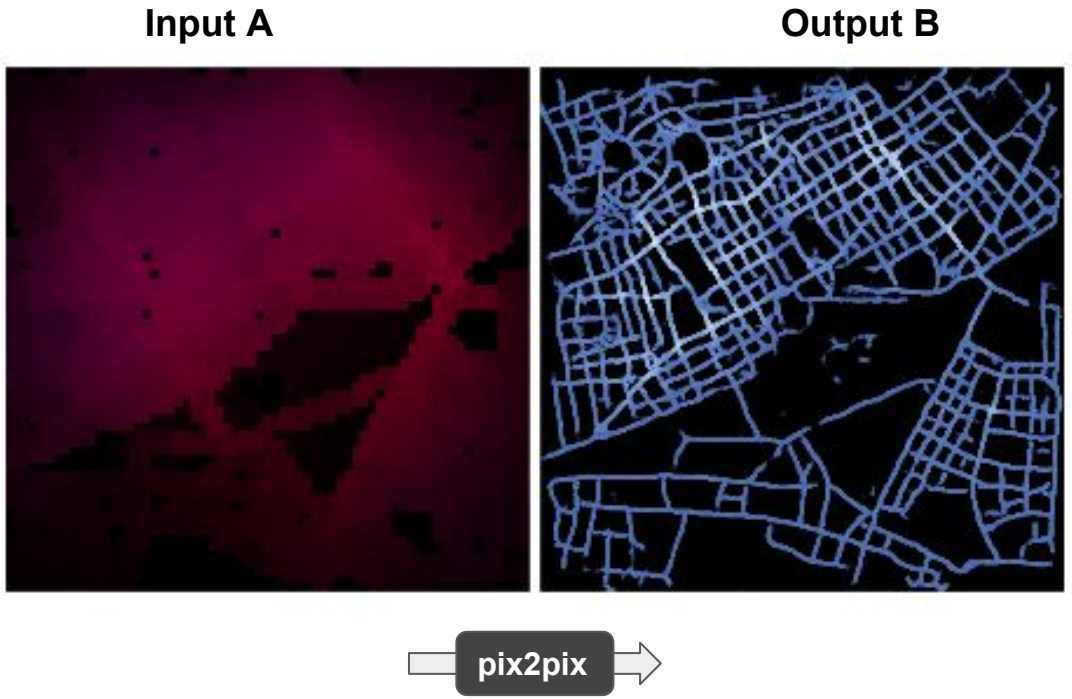
Workshop Topics and Tools

pix2pix - what does it do?

Generative Adversarial Networks (GAN)

Pix2pix

Generates new images based on an input image



Workshop Topics and Tools

pix2pix - what does it do?

Generative Adversarial Networks (GAN)

Pix2pix

Generates new images based on an input image

Workshop Topics and Tools

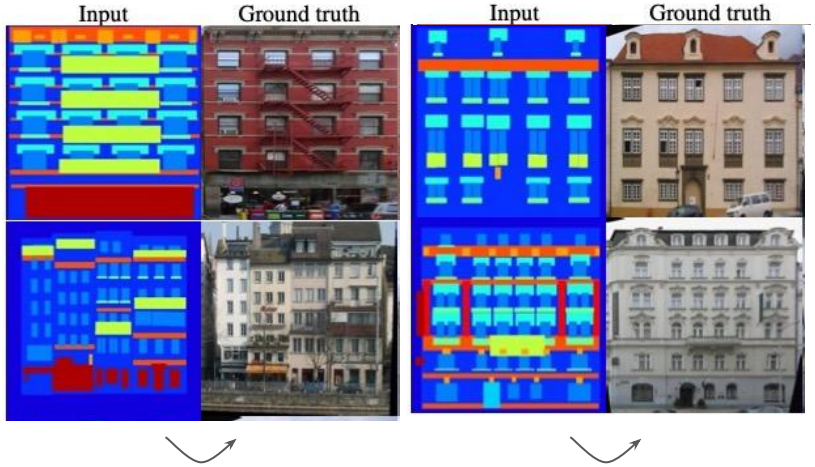
pix2pix - how does it do that?

Lots of training data!

With a few thousand image pairs:

A) A simplified or abstract representation of an image

B) The desired result image



Workshop Topics and Tools

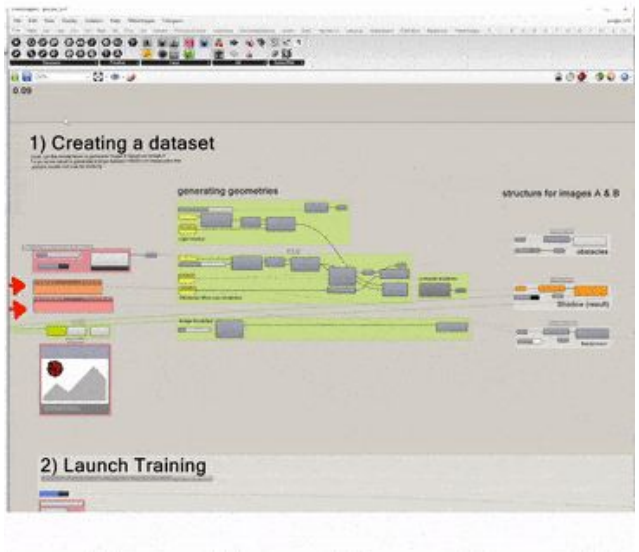
pix2pix - our grasshopper integration

We provide you a grasshopper definition that can:

A) Help you to create the training image pairs and bring them in the correct structure*

B) Trigger the training of a pix2pix model

C) Use your trained model within gh interactively



*you can also use the definition to load images from your drive and structure them into image pairs for training

Workshop Topics and Tools

pix2pix - our grasshopper integration

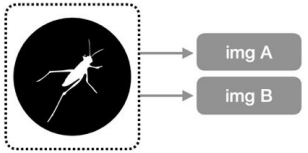
We provide you a grasshopper definition that can:

A) Help you to create the training image pairs and bring them in the correct structure*

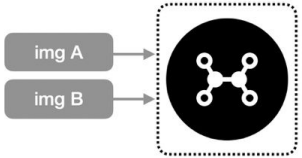
B) Trigger the training of a pix2pix model

C) Use your trained model within gh interactively

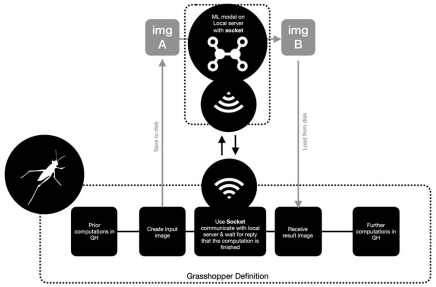
Create Training Data with Grasshopper



Train Model



Integrate with Grasshopper



*you can also use the definition to load images from your drive and structure them into image pairs for training

Workshop Topics and Tools

pix2pix

Example I

Problem:

I don’t want to count cars on satellite images manually anymore!

Workshop Topics and Tools

pix2pix

Example I

Use pix2pix !

1. **Prepare training data (image pairs)**
2. **Train model & test hyper parameters**
3. use model to detect cars on new satellite images

A - input

Original Image

B - real

Annotated image for training

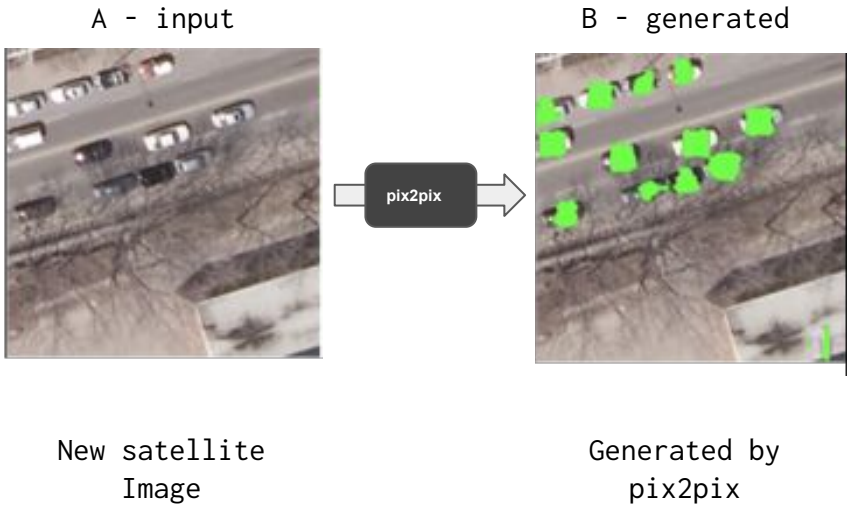
Workshop Topics and Tools

pix2pix

Example I

Use pix2pix !

1. Prepare training data (image pairs)
2. Train model & test hyper parameters
3. **use model to detect cars on new satellite images**



Workshop Topics and Tools

pix2pix

Example II

Problem:

I want a simple way to use my interactive parametric urban design model in the web and without being reliant on grasshopper!

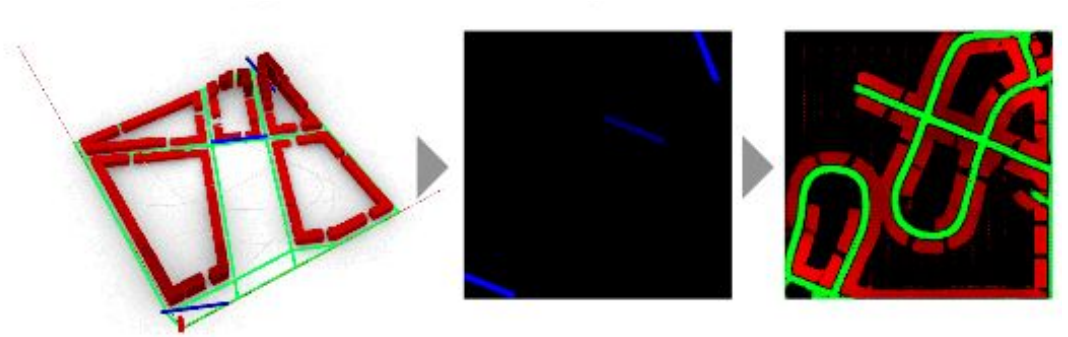
Workshop Topics and Tools

pix2pix

Example II

Use pix2pix !

1. **Prepare training data (image pairs)**
2. **Train model & test hyper parameters**
3. **use model to generate city model from three street segments**



Parametric Model **Image A** **Image B**

Image pairs for training
A → B

Workshop Topics and Tools

pix2pix

Example III

Problem:

I want to find the ideal location for a local waste-processing facility for each Block in the city. But using evolutionary optimization methods takes too much time.

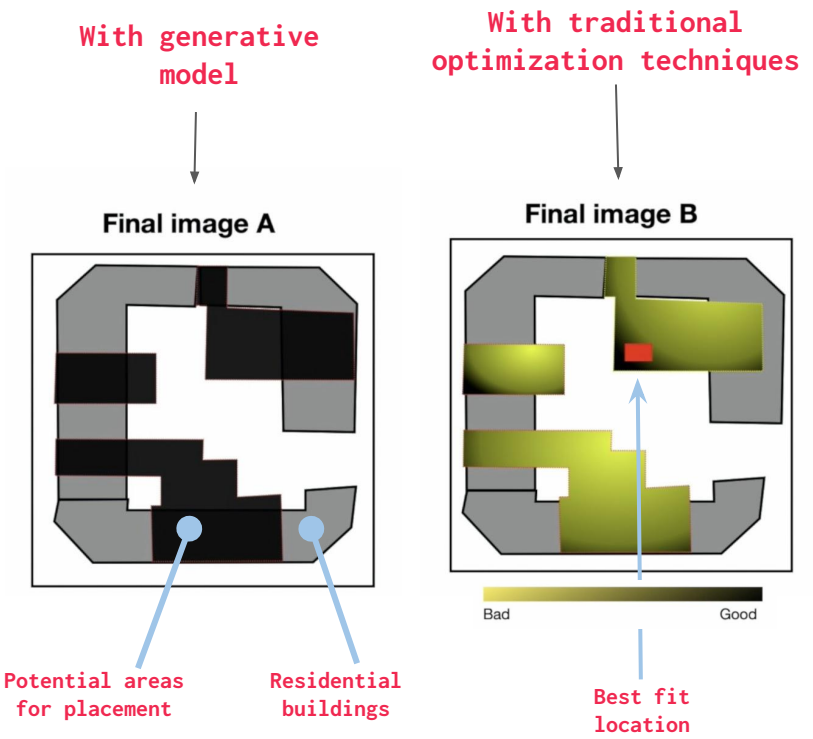
Workshop Topics and Tools

pix2pix

Example III

Use pix2pix !

1. **Prepare training data (image pairs) with traditional algorithms**
2. Train model & test hyper parameters
3. use model to place facility in one step



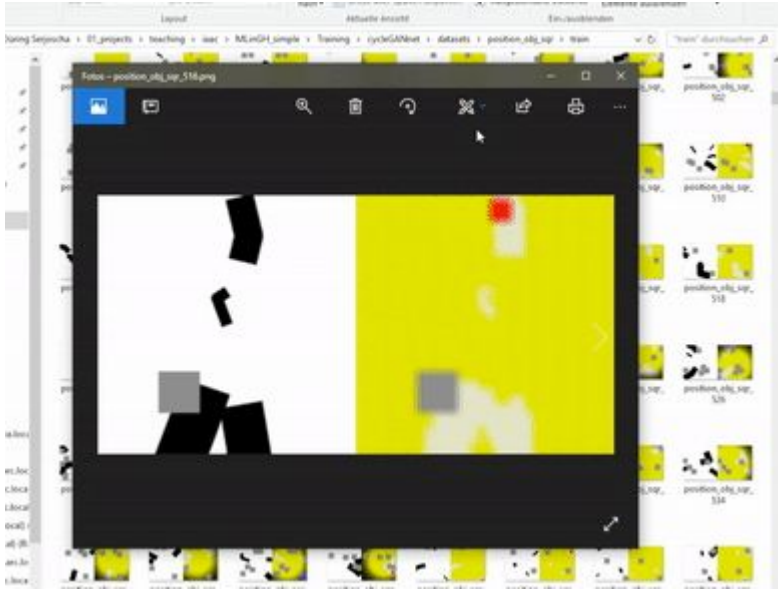
Workshop Topics and Tools

pix2pix

Example III

Use pix2pix !

1. **Prepare training data (image pairs) with traditional algorithms**
2. Train model & test hyper parameters
3. use model to place facility in one step



CIL CITY INTELLIGENCE LAB

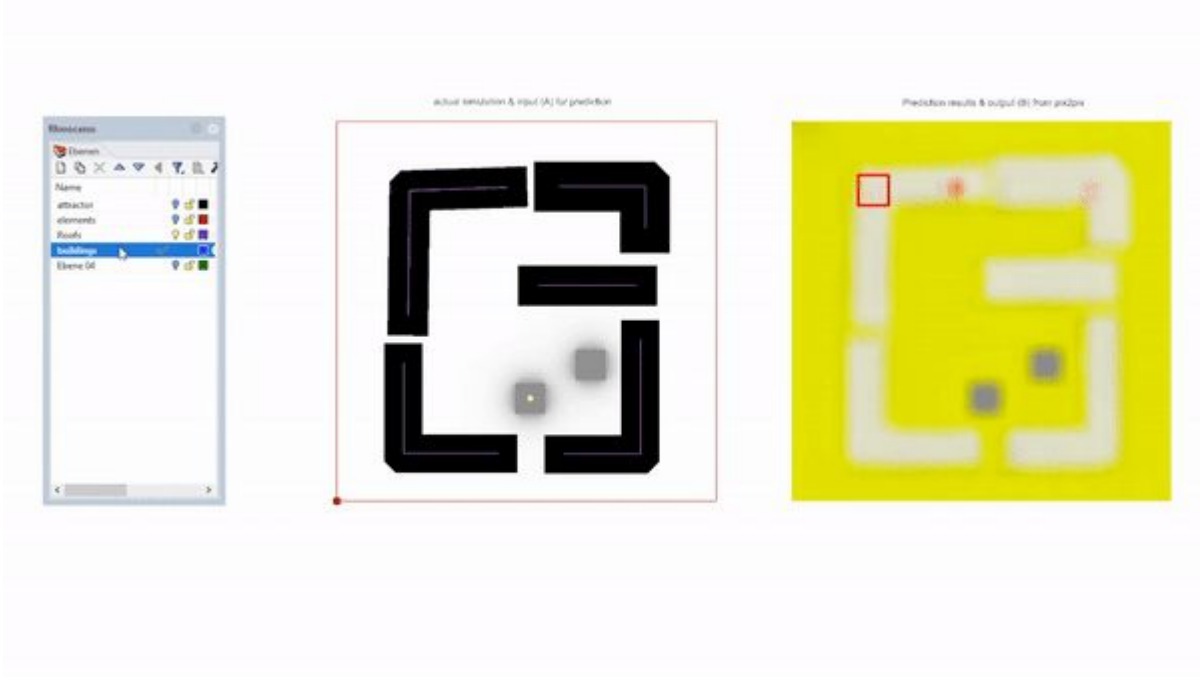
Workshop Topics and Tools

pix2pix

Example III

Use pix2pix !

1. Prepare training data (image pairs) with traditional algorithms
2. Train model & test hyper parameters
3. **use model to place facility in one step**



CIL CITY INTELLIGENCE LAB

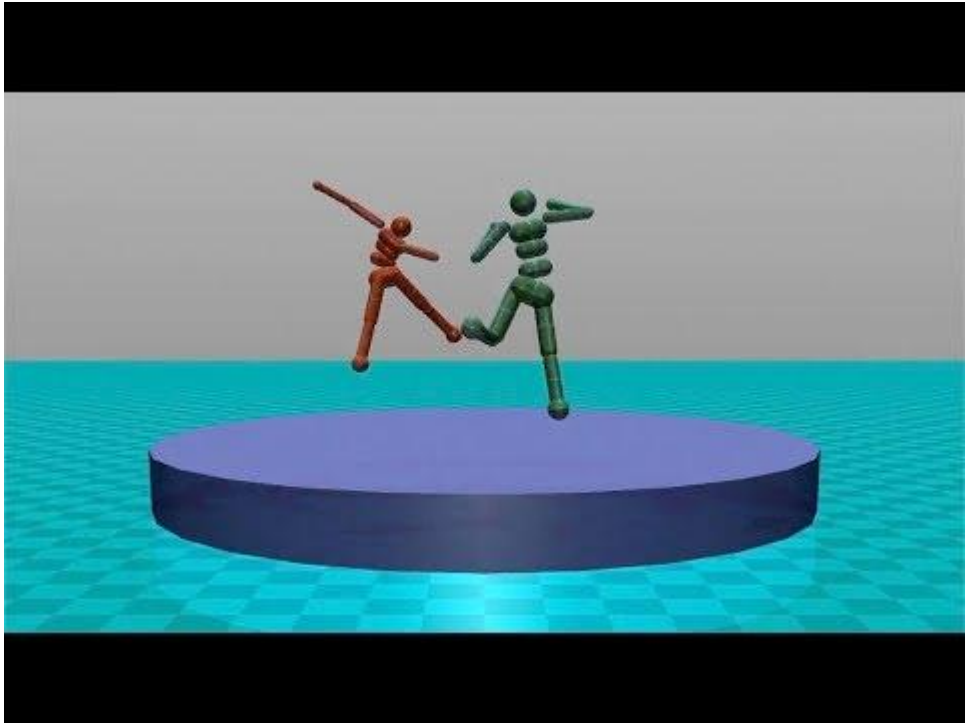
Workshop Topics and Tools

dql - what does it do?

Reinforcement learning (RL)

Deep Q-learning

Trains a decision making and strategy developing agent



Workshop Topics and Tools

dql - what does it do?

Reinforcement learning (RL)

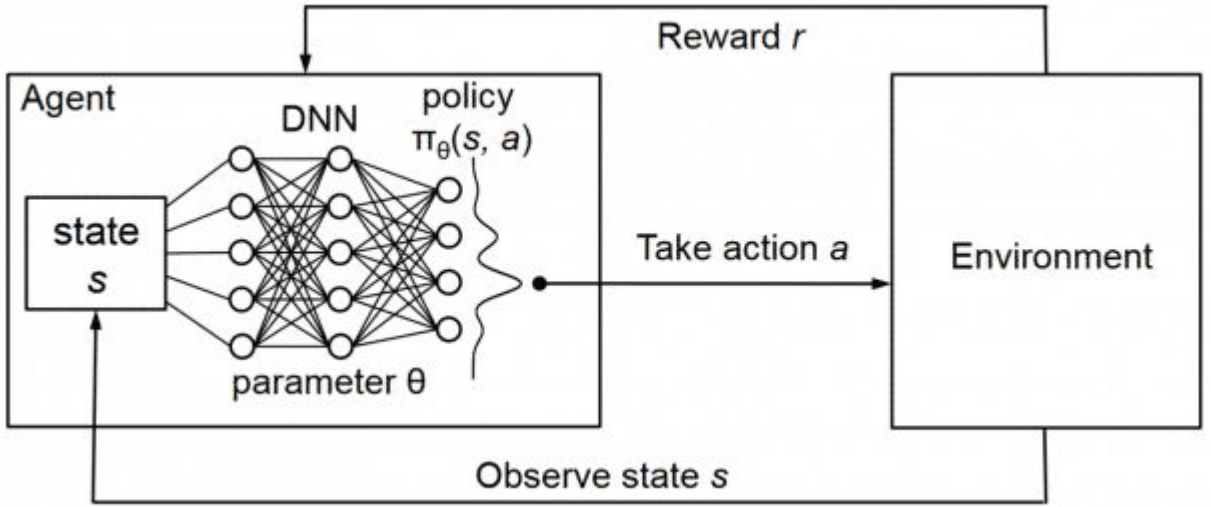
Deep Q-learning

Trains a decision making and strategy developing agent

CIL CITY INTELLIGENCE LAB

Workshop Topics and Tools

dql - how does it do that?



Workshop Topics and Tools

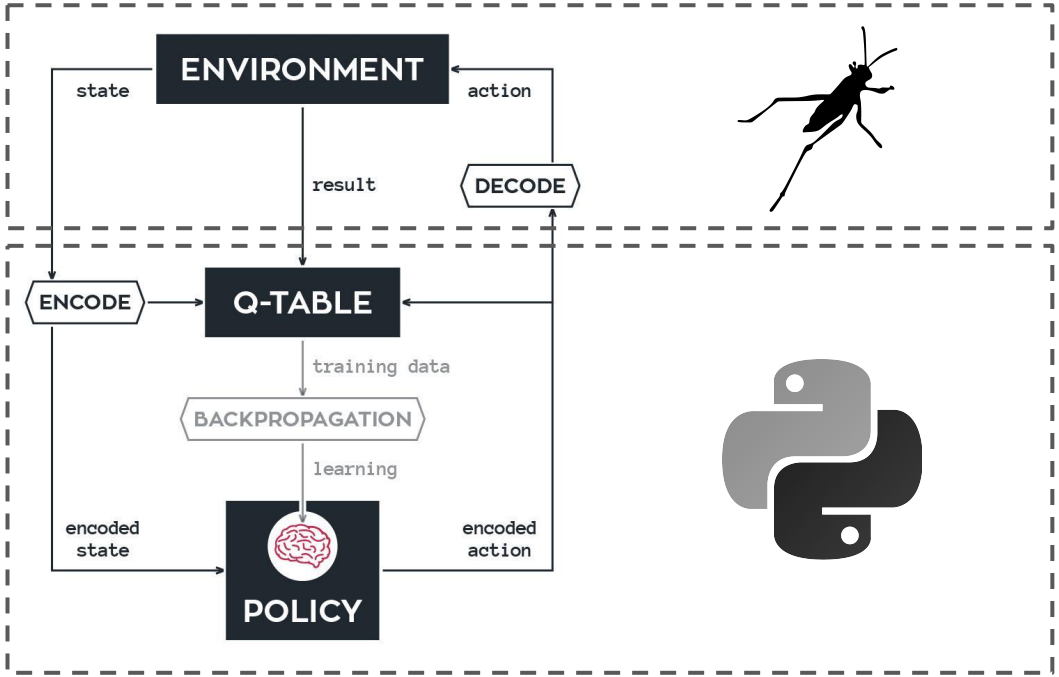
dqI - our grasshopper integration

We provide you a grasshopper definition that can:

A) Visualise and hold persistence over agent, state, action, and environment

B) Train a neural network through q-learning

C) Define a reward function within Grasshopper for more complex and even geometric functions



CIL CITY INTELLIGENCE LAB

Workshop Topics and Tools

dql

Example I

Problem:

I want to simulate the simple motion of an aimless car

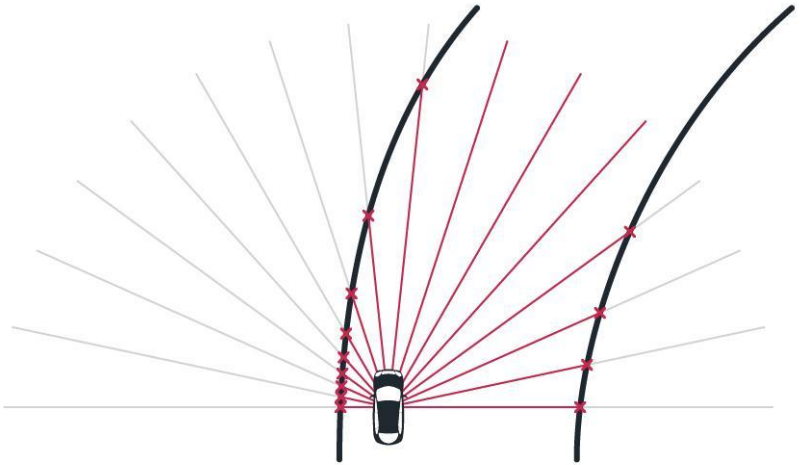
Workshop Topics and Tools

dql

Example I

Use dqn!

1. **Define agent, actions, and env in GH**
2. **Define a reward function**
3. Train model & test hyper parameters



[0.1248 , 0.1263 , 0.1335 , 0.1481 , 0.1739 , 0.2204 , 0.3103 , 0.4998 , 0.8425 , 1.0000 , 1.0000 , 1.0000 , 0.7750 , 0.6014 , 0.5272 , 0.4977]

CITY INTELLIGENCE LAB

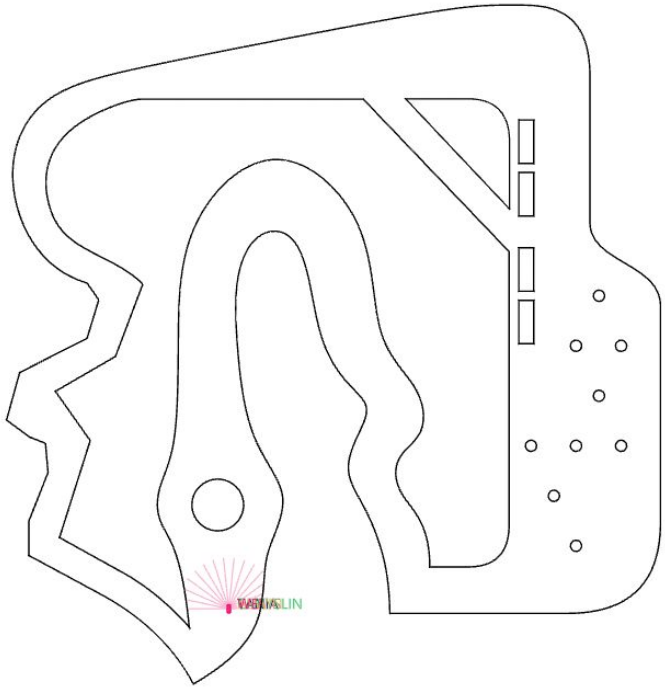
Workshop Topics and Tools

dql

Example I

Use dqn!

1. Define agent, actions, and env in GH
2. Define a reward function
3. **Train model & test hyper parameters**



CITY INTELLIGENCE LAB

Workshop Topics and Tools

Mask R-CNN

Mask R-CNN

Finds and labels objects in images aka. semantic segmentation.

Mask R-CNN

Kaiming He Georgia Gkioxari Piotr Dollár Ross Girshick

Facebook AI Research (FAIR)

Abstract

We present a conceptually simple, flexible, and general framework for object instance segmentation. Our approach efficiently detects objects in an image while simultaneously generating a high-quality segmentation mask for each instance. The method, called Mask R-CNN, extends Faster R-CNN by adding a branch for predicting an object mask in parallel *with the existing branch for bounding box recognition. Mask R-CNN is simple to train and adds only a small overhead to Faster R-CNN, running at 5 fps. Moreover,*

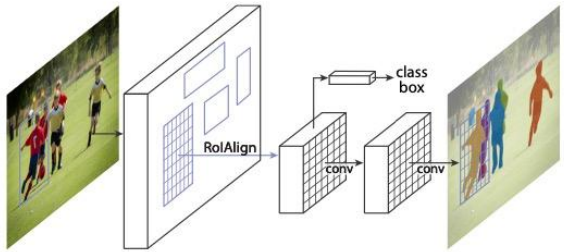


Figure 1. The **Mask R-CNN** framework for instance segmentation.

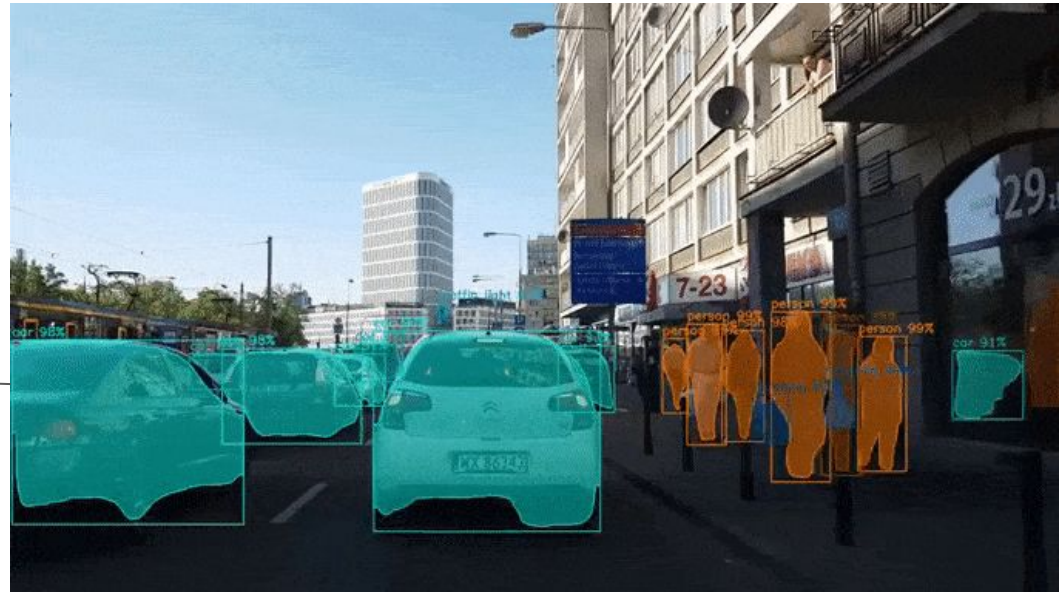
https://arxiv.org/pdf/1703.06870.pdf

Workshop Topics and Tools

Mask R-CNN - what does it do?

Mask R-CNN

Finds and labels objects in images aka. semantic segmentation.



CIL CITY INTELLIGENCE LAB

Workshop Topics and Tools

Mask R-CNN - our colab implementation

We prepared you a colab setup to use this model:

A) Pulled a repo with a DeepLabV3 implementation

B) Adjusted the source code for the colab environment

C) Allow multiple methods for own inferences

CIL CITY INTELLIGENCE LAB

Workshop Topics and Tools

Mask R-CNN

Example I

Problem:

I need a way to quantify the content of images of urban places in order to quickly find similar places eg. in terms of greenery, car traffic, pedestrian frequence or building heights.

Workshop Topics and Tools

Mask R-CNN

Example I

Use Mask R-CNN!

1. Use the pre-trained model “Cityscapes”
2. Scrap images of different places (street view images)
3. Run Mask R-CNN on these images and save results in a database
4. Query the database using the segmentation results of other images

Persons: 12
Urban Furniture: 33
Signs: 9
Cars: 3
Dogs: 0

Sky: 20%
Street: 40%
Sidewalk: 15%
Greenery: 15%
...

CIL CITY INTELLIGENCE LAB

Workshop Topics and Tools

Mask R-CNN

Example II

Problem:

I want to digitize a large amount of printed Floor plans automatically to use them within a BIM environment.

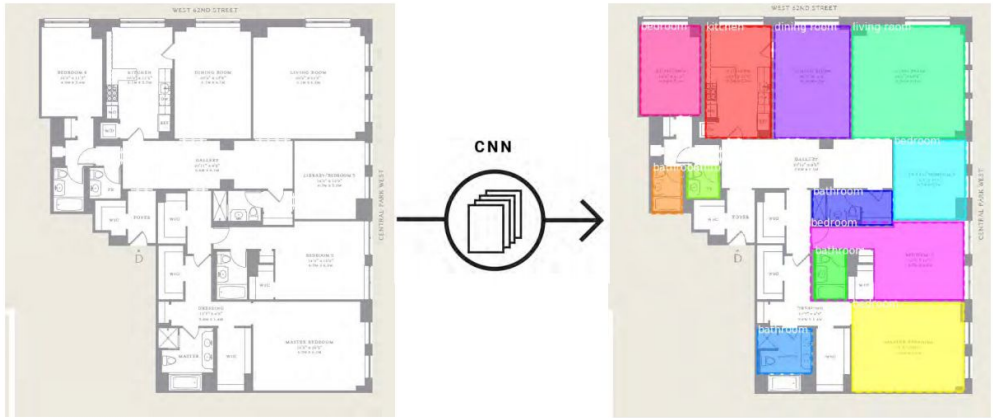
Workshop Topics and Tools

Mask R-CNN

Example II

Use Mask R-CNN!

1. Prepare training data (manually annotate a few hundred plans)
2. Train model
3. use model to semantically analyse floor plans, detect walls and different room categories.



CIL CITY INTELLIGENCE LAB

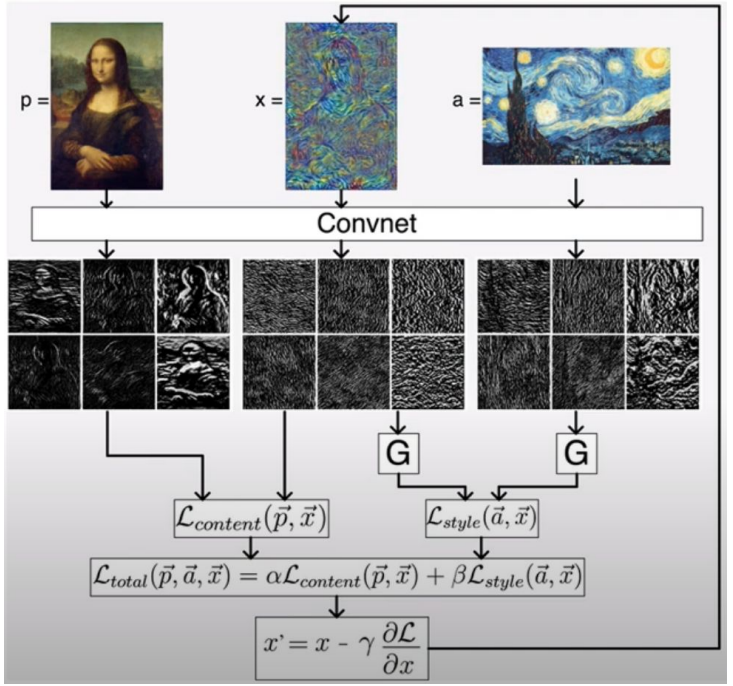
Workshop Topics and Tools

Style Transfer - what does it do?

Generative Adversarial Networks (GAN)

Style Transfer

Changes the look of images based on a reference image



CITY INTELLIGENCE LAB

Workshop Topics and Tools

Style Transfer - our colab implementation

We prepared you a colab setup to use this model:

A) Pulled a repo with a NVIDIA's Fast-Style-Transfer

B) Workflow for own inferences

Workshop Topics and Tools

Style Transfer

Example I

Problem:

I want to quickly stylise a 3D model.

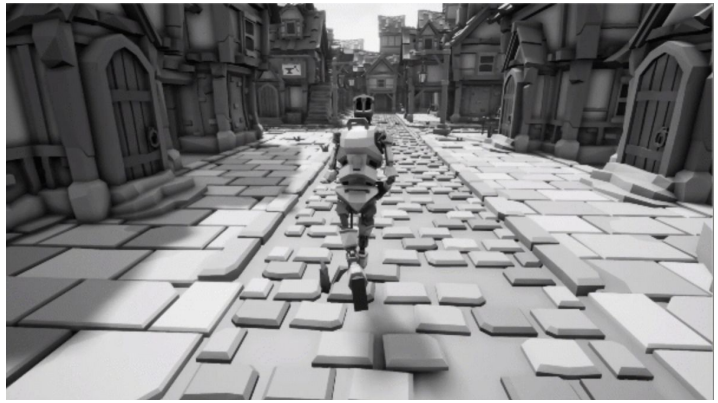
Workshop Topics and Tools

Style Transfer

Example I

Use Style Transfer!

1. Choose a style image
2. Apply that style to your content image



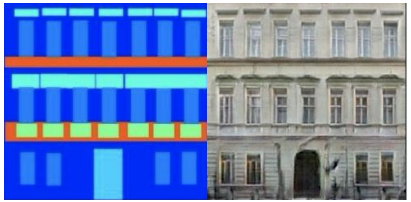
Workshop Topics and Tools

Machine learning models and workflows

Generative Adversarial Networks (GAN)

Pix2pix

Generates new images based on an input image

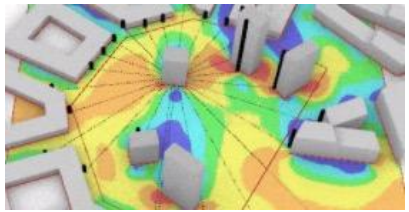


- Approximation of results of traditional optimization engines
- Generation of Buildings based on block outlines
- Rapid visualization of sketches (eg. substitute 3D render engines)

Reinforcement learning (RL)

Deep-Q-learning

Trains a decision making and strategy developing agent

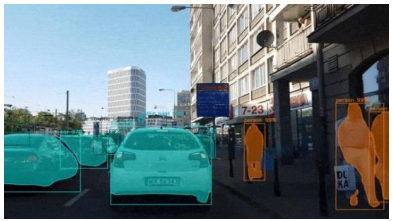


- Simulate crowd behavior with smart RL agents
- Simulate shortcut behavior of of pedestrians
- Manage and stabilize dynamic systems (eg. traffic lights and flows, lockdown policies)

Convolutional Neural Networks (CNN)

Mask R-CNN model

Finds and labels objects in images



- Quantify and understand the content of images
- Predicting the usage of bike sharing schemes and times of load.
- … use the results as training data for other models such as pix2pix or DQL-Agent training

Generative Adversarial Networks (GAN)

Style Transfer

Changes the look of images based on a reference image

- Advance visual quality of images
- Answer questions such as: how would this pace look like in winter, summer or during rain?

grasshopper (train + apply models)

google colab (apply pre-trained models)

Playground Model

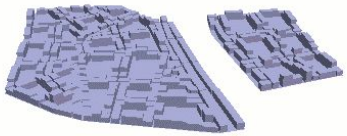
Location to apply the models

Location: Vienna

Data:
Street Network
Plots
Building Footprint + 3D
POI
+++

Playground Model

Example parametric model to generate cases



Typology Row

Building Height 12.0

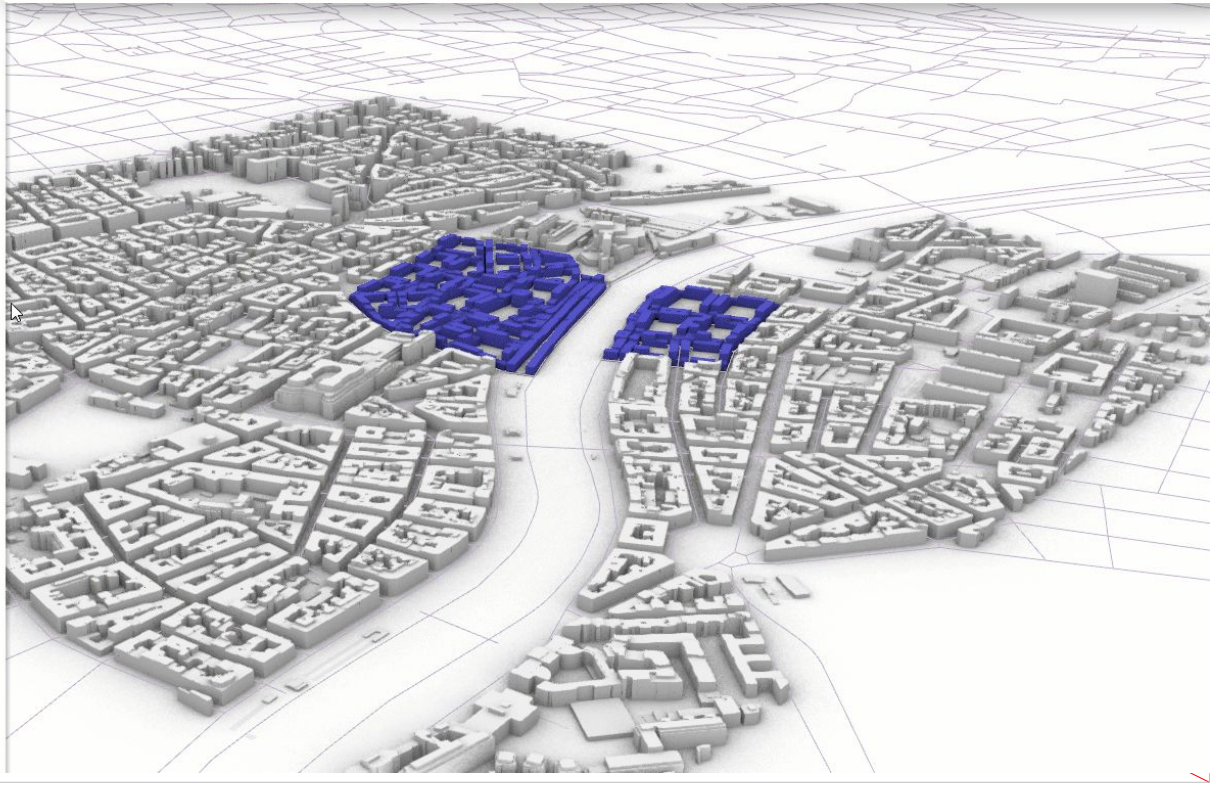
Diversity 40.0

Block Size 60

Parcel Width 14

Building Depth 13

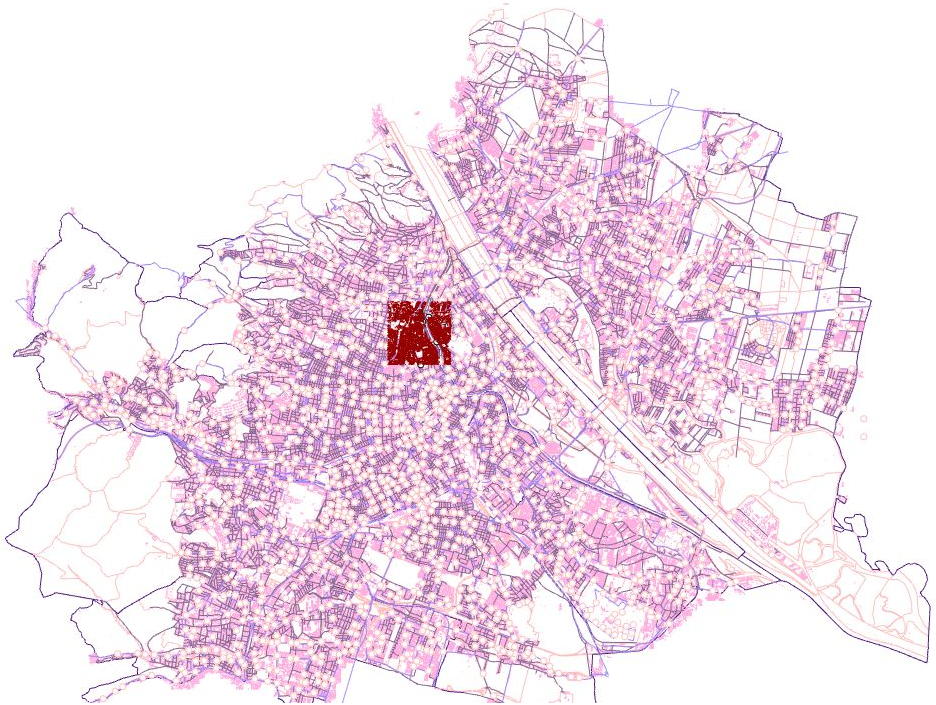
FAR 2.0



CIL CITY INTELLIGENCE LAB

Playground Model

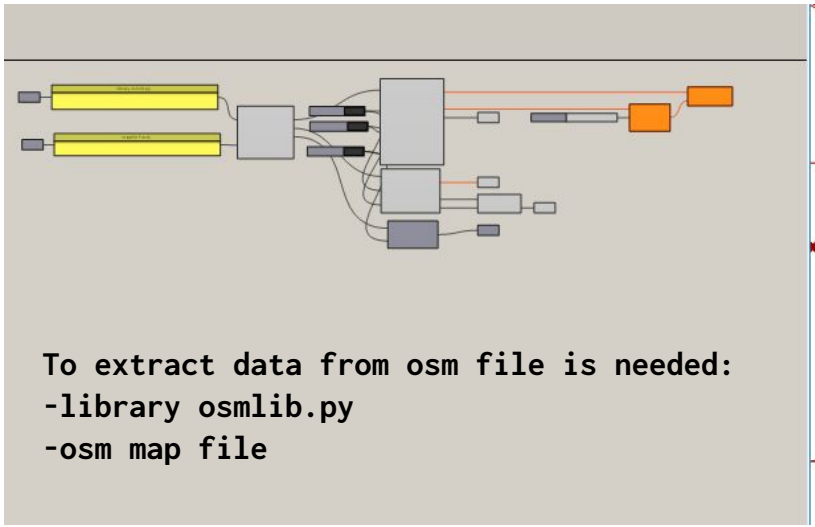
Shapefiles + gh definition



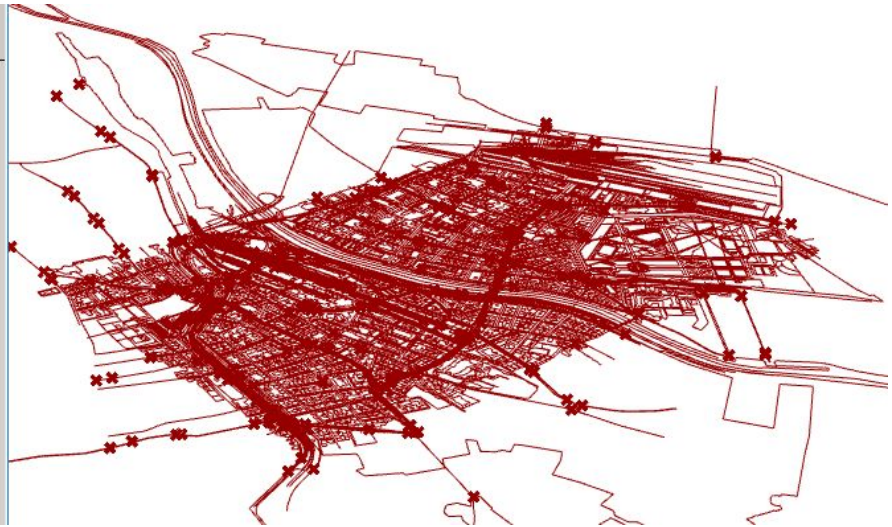
CIL CITY INTELLIGENCE LAB

Playground Model

OSM to Rhino



To extract data from osm file is needed:
-library osmlib.py
-osm map file



Potential Challenges

Formulating a project/challenge

Potential Challenges

A few examples

Main network and morphology labels

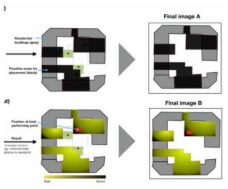
Full network and buildings

Project title

instantMasser

Problem

I want a simple way to use my interactive parametric urban design model in the web and without being reliant on grasshopper!* to create massing models.

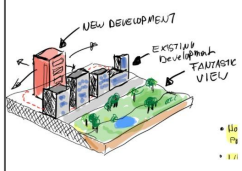


Project title

One-Step-Optimization

Problem

I want to find the ideal location for a local waste-processing facility for each city block. But using evolutionary optimization methods takes too much time and is complicated to set up.



Project title

AutoImo - smart building allocator

Problem

While evolutionary optimization can greatly improve design performance, it takes many iterations and long computing times, thus it's not the fit for human lead live design sessions.

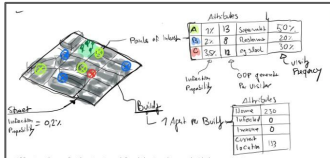


Illustration of the test model with Amenity and Citizen agent related attributes. Based on this system the DQN-Agent learns to make decisions.

Project title

Lockdown manager

Problem

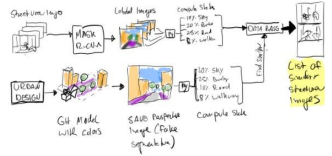
In times of pandemic it's crucial to decide when which kind of amenities need to shut-down to maintain a balance between healthcare and local economy. However, the non-linearity of the matter and time delays between policy measures and infection rates makes it hard for humans to make rational decisions.

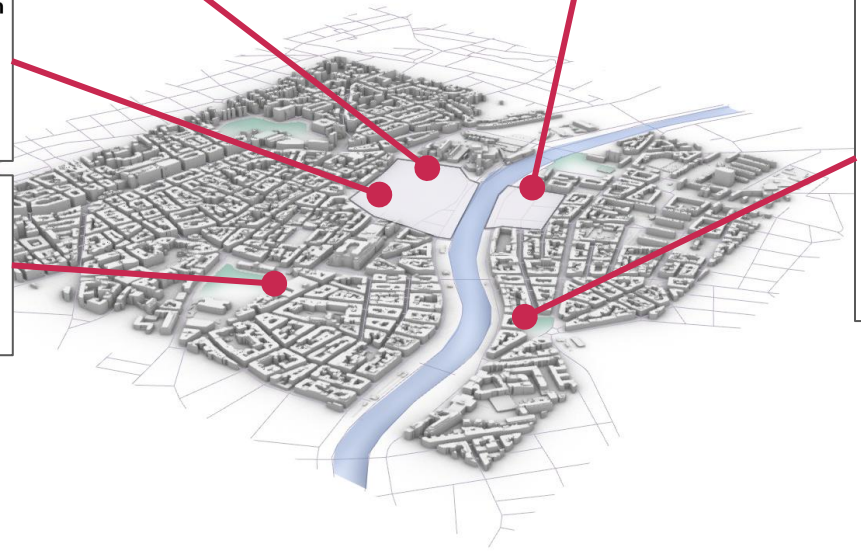
Project title

ReferenceRenderer

Problem

When rapidly creating many different massing options it is unfeasible to create detailed renders that try to imagine how a quarter would look and feel like from the human perspective.

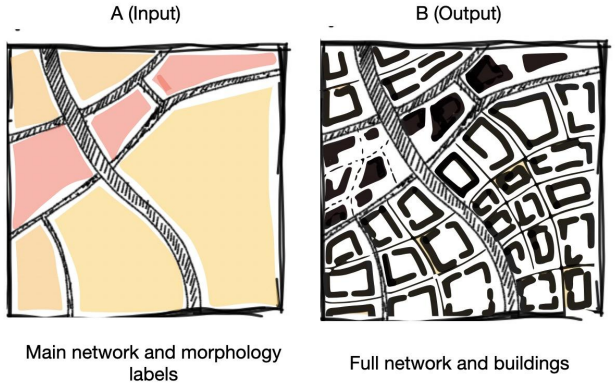




CIL CITY INTELLIGENCE LAB

Potential Challenges

Pix2pix 1



Project title

InstantMasser

Problem

I want a simple way to use my interactive parametric urban design model in the web and without being reliant on grasshopper!* to create massing models.

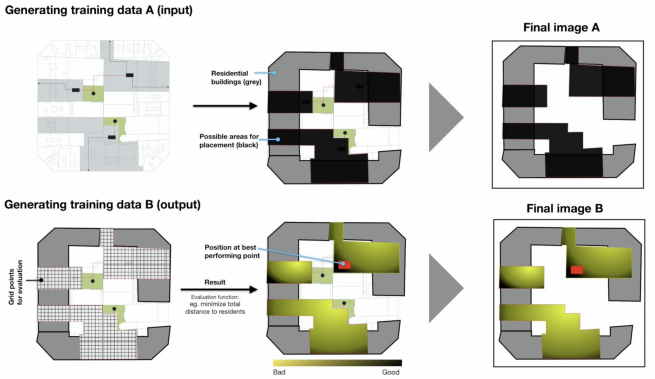
Solution

Our Hack uses pix2pix to learn how to represent the properties of a generative gh model based on a few input parameters aka a roughly sketched out map (image on the left). We encode different parameters of the generative model by colors (eg. red and yellow). This allows to quickly try out different variants and massing options.

*instead of parametric models, pix2pix could be trained with different typologies such as Barcelona or New York block etc.

Potential Challenges

Pix2pix 2



Illustrates process how to translate the problem into image spaces, pix2pix can work with. The final objective is to generate image B based on input A.

Project title

One-Step-Optimizer

Problem

I want to find the ideal location for a local waste-processing facility for each city block. But using evolutionary optimization methods takes too much time and is complicated to set up.

Solution

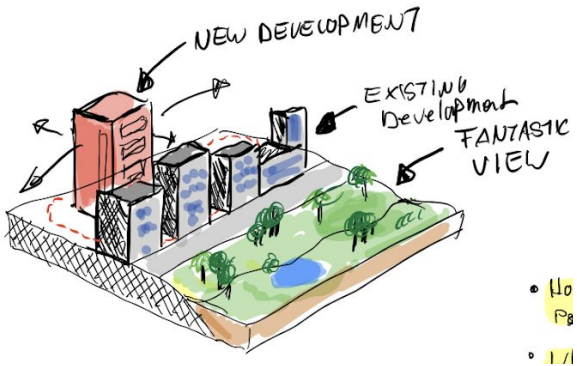
We train a pix2pix model on a diverse set of results generated by common evolutionary optimization. Once, trained we can conduct estimations of suitable locations in one step.

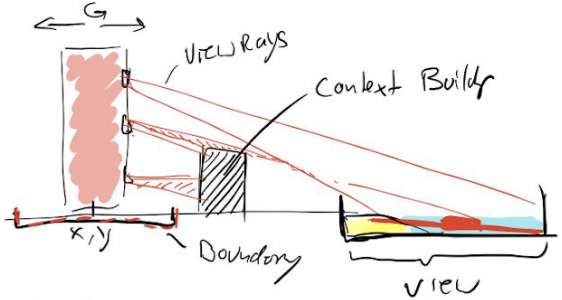
Implementation

Setup a generative model that creates variants to be evaluated by the traditional optimization algorithm. Structure data into input and output image, where the model should learn how to get to the output based on the input. Train model with a few thousand image pairs.
Apply to trained model to new cases.

CIL CITY INTELLIGENCE LAB

Potential Challenges
DQL 1





Project title
AutoImo - smart building allocator

Problem
While evolutionary optimization can greatly improve design performance, it takes many iterations and long computing times, thus it’s not the best fit for human lead live design sessions.

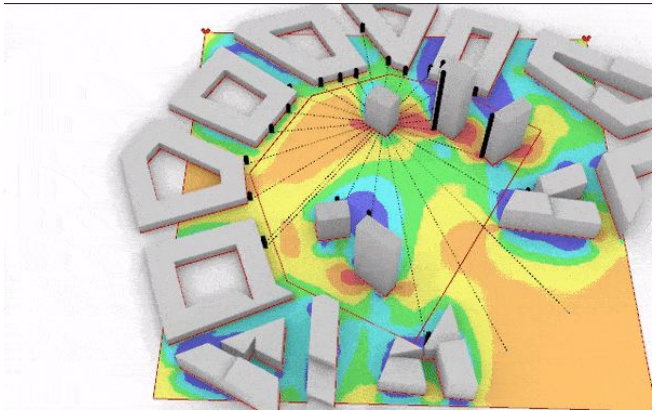
Solution
Our Hack uses a **DQN** to train a **Agent** (the building) to rapidly improve its location given a local environment. The system allows a designers intervention at anytime, allowing for high levels of interactivity.

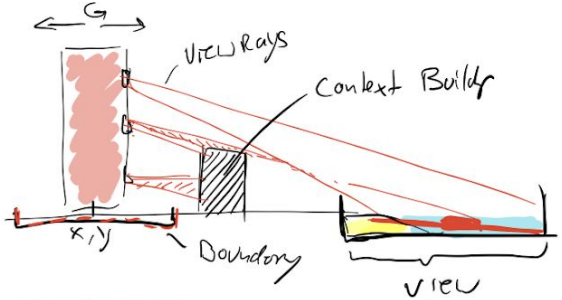
Implementation
Model: A parametric model is made in gh, that creates divers context (buildings, form of plot, area that provides a good view) for the training of the agent.
DQN: The DQN-Agent is represented by the building that a designer wants to add. For the prove-of-concept the agent tries to maximise the view on a nearby scenic-area while not breaking boundary conditions.

Environment: Location of building, location of context buildings, view factors
Agent’s actions: move forward, move backward, rotate left, rotate right
Reward function: max. view to scenic area, do not break boundary conditions

CITY INTELLIGENCE LAB

Potential Challenges
DQL 1





Project title
AutoImo - smart building allocator

Problem
While evolutionary optimization can greatly improve design performance, it takes many iterations and long computing times, thus it’s not the fit for human lead live design sessions.

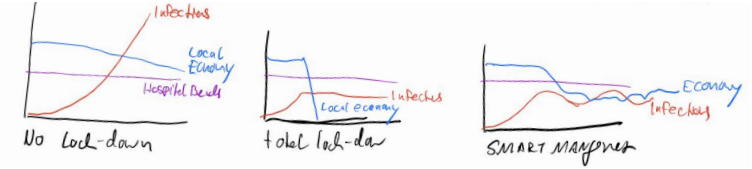
Solution
Our Hack uses a **DQN** to train a **Agent** (the building) to rapidly improve its location given a local environment. The system allows a designers intervention at anytime, allowing for high levels of interactivity.

Implementation
Model: A parametric model is made in gh, that creates divers context (buildings, form of plot, area that provides a good view) for the training of the agent.
DQN: The DQN-Agent is represented by the building that a designer wants to add. For the prove-of-concept the agent tries to maximise the view on a nearby scenic-area while not breaking boundary conditions.

Environment: Location of building, location of context buildings, view factors
Agent’s actions: move forward, move backward, rotate left, rotate right
Reward function: max. view to scenic area, do not break boundary conditions

CITY INTELLIGENCE LAB

Potential Challenges
DQL 2



Expected different trajectories of the system, from left to right: no shut-down measures are taken, tota shut-down, DQN-Agent regulated system. In the latter the system behaves more stable.

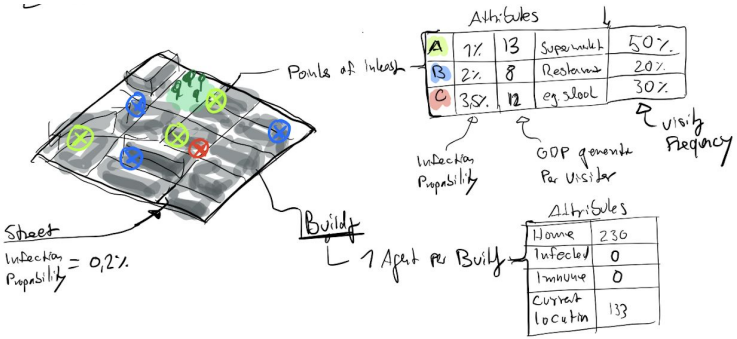


Illustration of the test model with Amenity and Citizen agent related attributes. Based on this system the DQN-Agent learns to make decisions.

Project title

Lockdown manager

Problem

In times of pandemic it's crucial to decide when which kind of amenities need to shut-down to maintain a balance between healthcare and local economy. However, the non-linearity of the matter and time delays between policy measures and infection rates makes it hard for humans to make rational decisions.

Solution

Our Hack uses a **DQN** to train a **Agent** to decide when which type of amenity should close in order to maintain the best overall performance.

Implementation

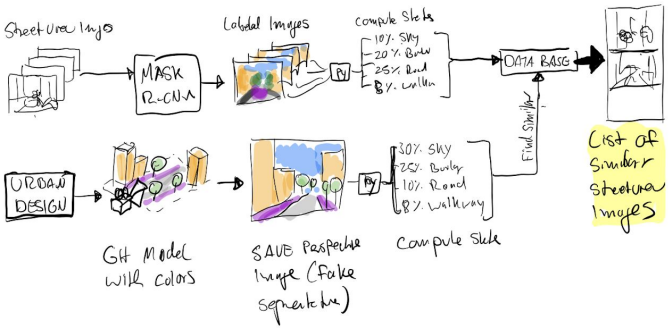
Model: A simple agent based (traditional, no DQN agent) approach is chosen to simulate citizens visiting different amenity types in their quarter (using the street-network). If an Amenity is shut-down they remain at home instead. The simulation starts with one person infected.
DQN: Is used to train an "agent" to make the decisions of when to cose certain amenity types.The model considers the following parameter:

Environment: Status of amenities (open/closed), Infection rate, local economy, hospital capacity.
Agent's actions: close or reopen types of amenities
Reward function: min. infections, max. local economy, stay below hospital capacity

CIL CITY INTELLIGENCE LAB

Potential Challenges

Mask R-CNN



Top row: Mask R-CNN workflow of processing street view images, and save their semantic features to a database. Bottom: Workflow to create semantic maps with grasshopper. Objects such as buildings, streets or trees are colored in the same color scheme used in Mask R-CNN.

Project title

ReferenceRenderer

Problem

When rapidly creating many different massing options it is unfeasible to create detailed renders that try to imagine how a quarter would look and feel like from the human perspective.

Solution

First, we build up a database of street view images, semantically analysed by Mask R-CNN. For each massing model we then generate a simple perspective with the same color schema used for R-CNN. We then query our database for semantic similar places and display the top five matches.

CIL CITY INTELLIGENCE LAB

Potential Challenges

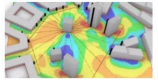
A few examples

Generative Adversarial Networks (GAN)
Pix2pix

Generates new images based on an input image

Reinforcement learning (RL)
Deep-Q-learning

Trains a decision making and strategy developing agent

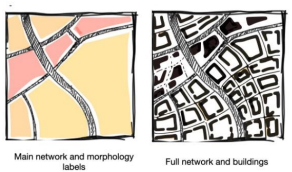


Convolutional Neural Networks (CNN)
Mask R-CNN model

Finds and labels objects in images

Generative Adversarial Networks (GAN)
Style Transfer

Changes the look of images based on a reference image

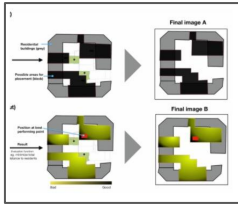


Project title

instantMasser

Problem

I want a simple way to use my interactive parametric urban design model in the web and without being reliant on grasshopper!* to create massing models.

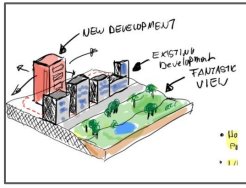


Project title

One-Step-Optimization

Problem

I want to find the ideal location for a local waste-processing facility for each city block. But using evolutionary optimization methods takes too much time and is complicated to set up.

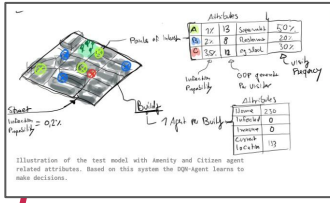


Project title

AutoImo - smart building allocator

Problem

While evolutionary optimization can greatly improve design performance, it takes many iterations and long computing times, thus it's not the fit for human lead live design sessions.



Project title

Lockdown manager

Problem

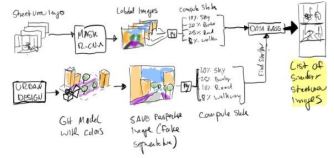
In times of pandemic it's crucial to decide when which kind of amenities need to shut-down to maintain a balance between healthcare and local economy. However, the non-linearity of the matter and time delays between policy measures and infection rates makes it hard for humans to make rational decisions.

Project title

ReferenceRenderer

Problem

When rapidly creating many different massing options it is unfeasible to create detailed renders that try to imagine how a quarter would look and feel like from the human perspective.

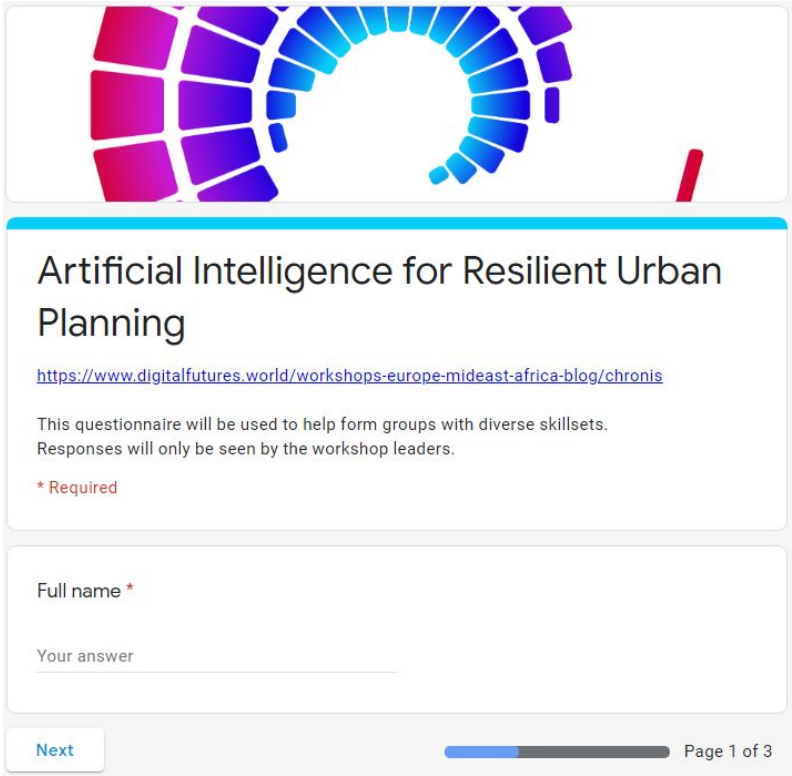


CIL CITY INTELLIGENCE LAB

Group Forming

Algorithmic Groups

BETTER THAN RANDOM(?)



Artificial Intelligence for Resilient Urban Planning

https://www.digitalfutures.world/workshops-europe-mideast-africa-blog/chronis

This questionnaire will be used to help form groups with diverse skillsets.
Responses will only be seen by the workshop leaders.

* Required

Full name *

Your answer

Next

Page 1 of 3

CIL CITY INTELLIGENCE LAB

Algorithmic Groups

BETTER THAN RANDOM(?)

Timestamp	Full name	Language-based	Visual programm	Artificial intelliger	Urban planning c	Building vs urbar	Parametric desig	Generative desig	Performance opt	Spatial analysis	Mobility and Trar	Interactive interfa
6/25/2020 12:41:38	Efa	5	10	1	10	10	10	10	10	10	2	10
6/25/2020 12:59:25	Sébastien Perrault	0	10	5	5	10	10	10	10	10	10	10
6/25/2020 14:20:55	Zvonko Vugreshek	5	9	8	7	6	9	9	8	8	7	7
6/25/2020 14:39:10	Prianka Bali	1	2	0	7	10	8	8	7	9	5	10
6/25/2020 17:29:32	Wanrongmiao "Mila" Zhar	1	2	9	10	4	1	5	1	6	10	10
6/25/2020 19:05:11	Stepan Kukharskiy	5	10	2	7	8	10	10	10	10	10	10
6/25/2020 20:17:22	Lea Khairallah	4	9	3	9	10	10	10	10	10	5	10
6/25/2020 21:32:33	Sahar Naz TALEB NEZHA	0	1	0	5	10	2	5	6	7	7	7
6/25/2020 22:05:22	Michael Thomas Walsh	6	5	4	4	5	8	8	8	7	7	7
6/26/2020 9:03:18	Ekaterina Vititneva	2	5	1	7	10	10	10	10	10	10	10
6/26/2020 11:04:38	Sanjay Somanath	4	9	5	7	5	8	8	8	10	3	10
6/26/2020 12:07:11	christina doumpioti	3	8	4	7	8	8	8	8	8	2	5
6/26/2020 12:36:35	yunyang ma	3	5	2	3	6	6	7	7	7	6	6
6/26/2020 13:22:19	Kr Pueblos	3	6	1	9	10	9	8	6	9	8	6
6/26/2020 14:29:28	José Aderson Araújo Pas	6	10	6	7	10	10	10	10	10	7	7
6/26/2020 14:54:56	Effie	7	9	6	6	5	10	10	9	9	7	10
6/26/2020 19:10:25	Anastasia Nikitsina	2	4	0	6	7	9	9	6	7	6	6
6/27/2020 0:24:56	Efa	10	10	5	10	10	10	10	10	9	5	10
6/27/2020 9:48:06	Nour Alkhaja	5	9	8	9	8	8	8	7	6	4	4
6/27/2020 10:21:33	Francesco De Luca	1	8	1	7	10	9	7	8	7	5	7
6/27/2020 12:22:48	Artem Gilmanov	5	7	2	7	9	6	7	9	8	3	5
6/27/2020 13:19:08	Javier Argota Sanchez-Va	7	8	7	10	10	8	10	8	10	10	8
6/27/2020 17:10:21	Eduard Petriu	8	8	5	5	9	10	10	10	7	5	9

CIL CITY INTELLIGENCE LAB

Algorithmic Groups
BETTER THAN RANDOM(?)

Timestamp	Full name	Language-based	Visual programm	Artificial intelliger	Urban planning	Building vs urbar	Parametric desig	Generative desig	Performance opt	Spatial analysis	Mobility and Trar	Interactive interfa
6/25/2020 12:41:38	Efa	5	10	1	10	10	10	10	10	10	2	10
6/25/2020 12:59:25	Sébastien Perrault	0	10	5	5	10	10	10	10	10	10	10
6/25/2020 14:20:55	Zvonko Vugreshek	5	9	8	7	6	9	9	8	8	7	7
6/25/2020 14:39:10	Prianka Bali	1	2	0	7	10	8	8	7	9	5	10
6/25/2020 17:29:32	Wanrongmiao "Mila" Zhar	1	2	9	10	4	1	5	1	6	10	10
6/25/2020 19:05:11	Stepan Kukharskiy	5	10	2	7	8	10	10	10	10	10	10
6/25/2020 20:17:22	Lea Khairallah	4	9	3	9	10	10	10	10	10	5	10
6/25/2020 21:32:33	Sahar Naz TALEB NEZHA	0	1	0	5	10	2	5	6	7	7	7
6/25/2020 22:05:22	Michael Thomas Walsh	6	5	4	4	5	8	8	8	7	7	7
6/26/2020 9:03:18	Ekaterina Vititneva	2	5	1	7	10	10	10	10	10	10	10
6/26/2020 11:04:38	Sanjay Somanath	4	9	5	7	5	8	8	8	10	3	10
6/26/2020 12:07:11	christina doumpioti	3	8	4	7	8	8	8	8	8	2	5
6/26/2020 12:36:35	yunyang ma	3	5	2	3	6	6	7	7	7	6	6
6/26/2020 13:22:19	Kr Pueblos	3	6	1	9	10	9	8	6	9	8	6
6/26/2020 14:29:28	José Aderson Araújo Pas	6	10	6	7	10	10	10	10	10	7	7
6/26/2020 14:54:56	Effie	7	9	6	6	5	10	10	9	9	7	10
6/26/2020 19:10:25	Anastasia Nikitsina	2	4	0	6	7	9	9	6	7	6	6
6/27/2020 0:24:56	Efa	10	10	5	10	10	10	10	10	9	5	10
6/27/2020 9:48:06	Nour Alkhaja	5	9	8	9	8	8	8	7	6	4	4
6/27/2020 10:21:33	Francesco De Luca	1	8	1	7	10	9	7	8	7	5	7
6/27/2020 12:22:48	Artem Gilmanov	5	7	2	7	9	6	7	9	8	3	5
6/27/2020 13:19:08	Javier Argota Sanchez-Va	7	8	7	10	10	8	10	8	10	10	8
6/27/2020 17:10:21	Eduard Petriu	8	8	5	5	9	10	10	10	7	5	9

CIL CITY INTELLIGENCE LAB

Algorithmic Groups

BETTER THAN RANDOM(?)

```
[ ]  from google.colab import auth
     from oauth2client.client import GoogleCredentials
     import gspread

     auth.authenticate_user()
     gc = gspread.authorize(GoogleCredentials.get_application_default())
```

k-means clustering

From Wikipedia, the free encyclopedia

Not to be confused with K-nearest neighbors algorithm.

***k*-means clustering** is a method of vector quantization, originally from signal processing, that aims to partition *n* observations into *k* clusters in which each observation belongs to the cluster with the nearest mean (cluster centers or cluster centroid), serving as a prototype of the cluster. This results in a partitioning of the data space into Voronoi cells. It is popular for cluster analysis in data mining. *k*-means clustering minimizes within-cluster variances (squared Euclidean distances), but not regular Euclidean distances, which would be the more difficult Weber problem: the mean optimizes squared errors, whereas only the geometric median minimizes Euclidean distances. For instance, better Euclidean solutions can be found using k-medians and k-medoids.

Forming Groups

t-distributed stochastic neighbor embedding

From Wikipedia, the free encyclopedia

"TSNE" redirects here. For the Boston-based organization, see Third Sector New England.

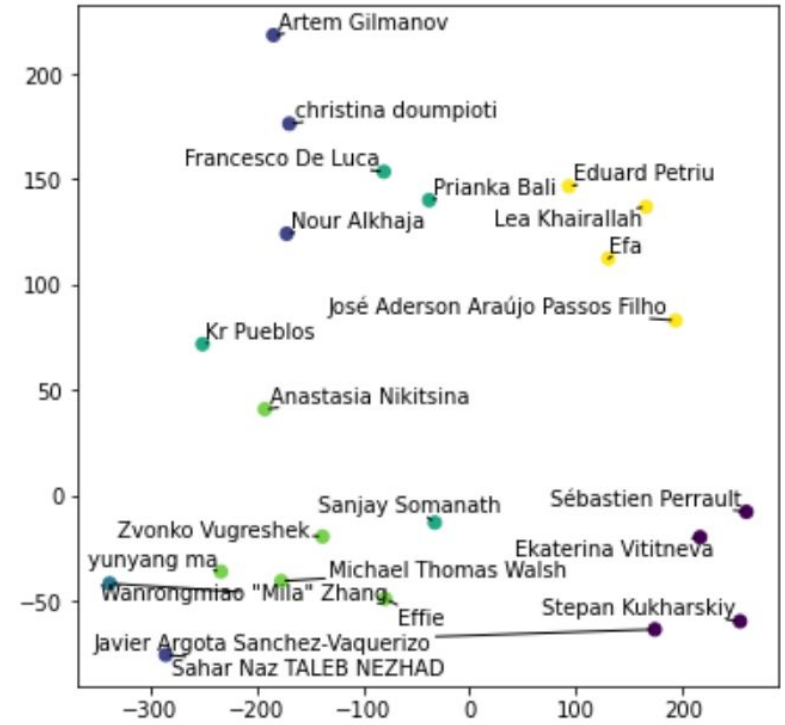
T-distributed Stochastic Neighbor Embedding (t-SNE) is a machine learning algorithm for visualization developed by Laurens van der Maaten and Geoffrey Hinton.[1] It is a nonlinear dimensionality reduction technique well-suited for embedding high-dimensional data for visualization in a low-dimensional space of two or three dimensions. Specifically, it models each high-dimensional object by a two- or three-dimensional point in such a way that similar objects are modeled by nearby points and dissimilar objects are modeled by distant points with high probability.

Visualisation

Algorithmic Groups

BETTER THAN RANDOM(?)

t-SNE ran for 1699 iterations.



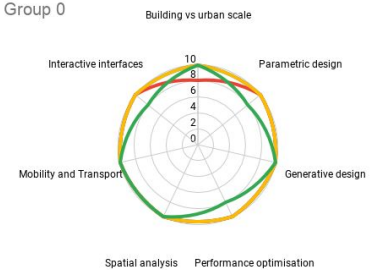

```
Group 0:
  - Sébastien Perrault
  - Stepan Kukharskiy
  - Ekaterina Vititneva
  - Javier Argota Sanchez-Vaquerizo
Group 1:
  - Sahar Naz TALEB NEZHAD
  - christina doumpioti
  - Nour Alkhaja
  - Artem Gilmanov
Group 2:
  - Wanrongmiao "Mila" Zhang
Group 3:
  - Prianka Bali
  - Sanjay Somanath
  - Kr Pueblos
  - Francesco De Luca
Group 4:
  - Zvonko Vugreshek
  - Michael Thomas Walsh
  - yunyang ma
  - Effie
  - Anastasia Nikitsina
Group 5:
  - Lea Khairallah
  - José Aderson Araújo Passos Filho
  - Efa
  - Eduard Petriu
```

CIL CITY INTELLIGENCE LAB

Algorithmic Groups

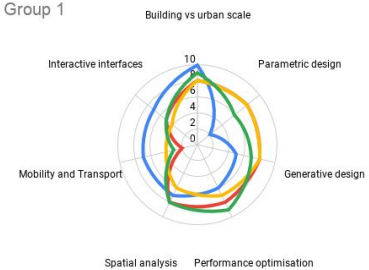
BETTER THAN RANDOM(?)

Group 0



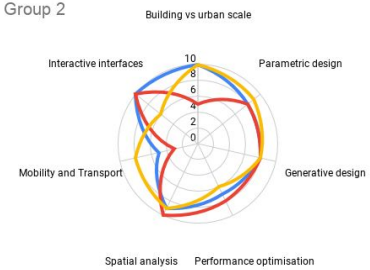
“Team Keen Beans”

Group 1



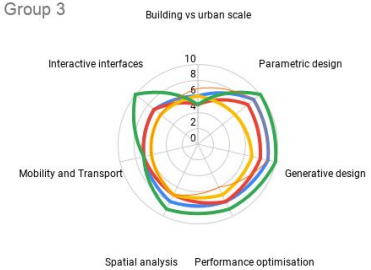
“Scale Squad”

Group 2



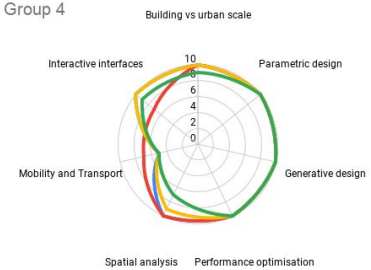
“Team Spatial Analysis”

Group 3



“Parametric Collective”

Group 4



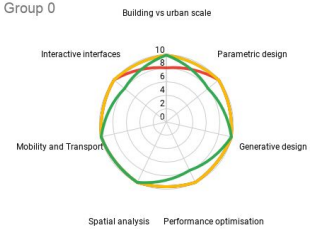
“Team Really Anti-Mobility”

CITY INTELLIGENCE LAB

Algorithmic Groups

BETTER THAN RANDOM(?)

Group 0



"Team Keen Beans"

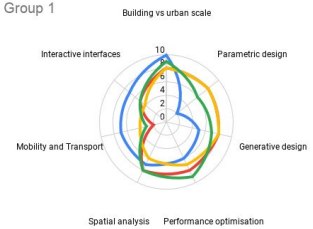
Sebastien Perrault

Stepan Kukharskiy

Ekaterina Vititneva

Javier Argota Sanchez-Vaquerizo

Group 1



"Scale Squad"

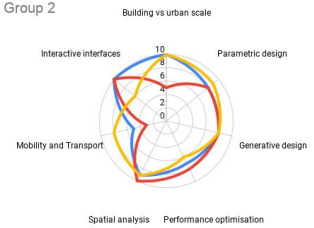
Sahar Naz Taleb Nezhad

Christina Doumpioti

Nour Alkhaja

Artem Gilmanov

Group 2



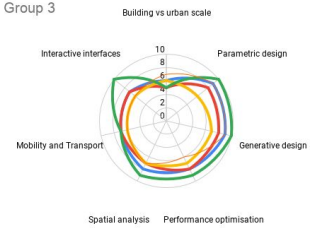
"Team Spatial Analysis"

Prianka Bali

Sanjay Somanath

Kr Pueblos

Group 3



"Parametric Collective"

Zvonko Vugreshek

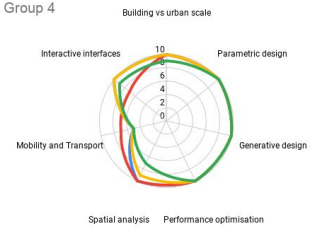
Michael Thomas Walsh

Yunyang Ma

Effie Douroudi

Anastasia Nikitsina

Group 4



"Team Really Anti-Mobility"

Lea Khairallah

José Aderson Araújo Passos Filho

Efa Ahvazi

Eduard Petriu

CITY INTELLIGENCE LAB

Outlook and day two

Day 2 (Mon 29th)

Name	Room	Time (GMT)	Description
Talk: Intro	Webinar	10:00 - 10:05	Recap + today's schedule
Talk: Intro to Computer Vision	Webinar	10:05 - 10:20	How do computers make sense out of images using conventional. neural networks?
Demo: Semantic Segmentation	Webinar	10:20 - 10:35	Introduction to semantic segmentation in Colab and how to use it on your own images
Demo: Style Transfer	Webinar	10:35 - 10:50	Introduction to style transfer in Colab and how to use it on your own images
Demo: pix2pix in GH	Webinar	10;50 - 11:20	Introduction of the pix2pix model and demo on how to integrate it into GH workflow
Demo: DQL in GH	Webinar	11:20 - 11:50	Introduction of the DQL model and demo on how to integrate it into GH workflow
Talk: Summary	Webinar	11:50 - 12:00	Overview and summary of models
Break	break	12:00 - 12:15	-
Exercise: Project Work	Meeting	12:15 - 13:30	In breakout rooms with advisory supervision
Presentation: Refined Pitches	Meeting	13:30 - 14:00	Update presentation from groups

CITY INTELLIGENCE LAB