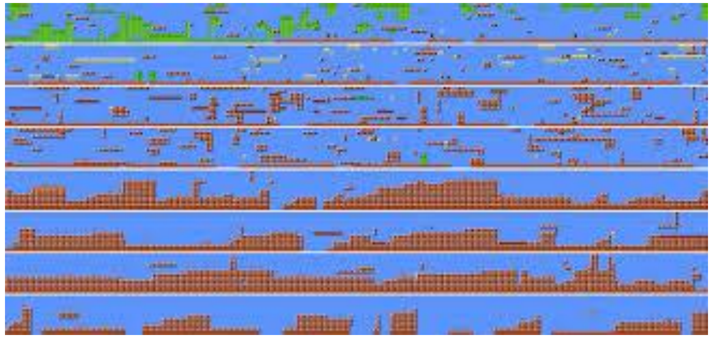
tile2tile:
Learning Game Filters for Platformer Style Transfer

Anurag Sarkar and Seth Cooper

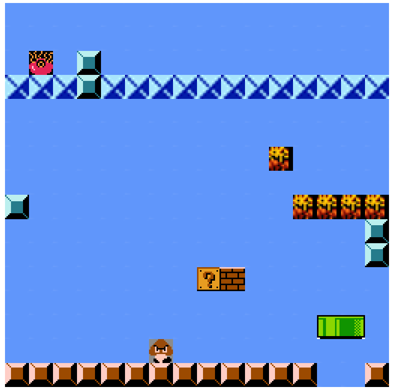
Northeastern University

Motivation

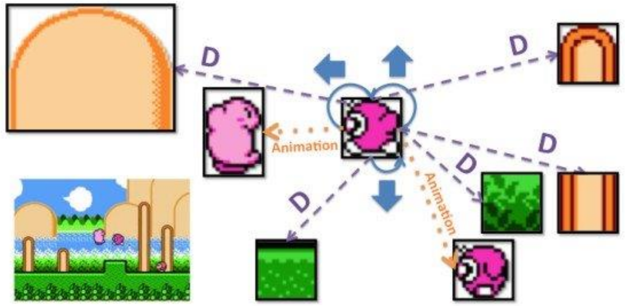
- Recent PCGML research has focused on learning and reasoning about design spaces spanning a number of games:
  - Generating new games by recombining learned game graphs
  - Blending latent representations
  - Affordance-based tile embeddings across multiple games
  - Domain adaptation



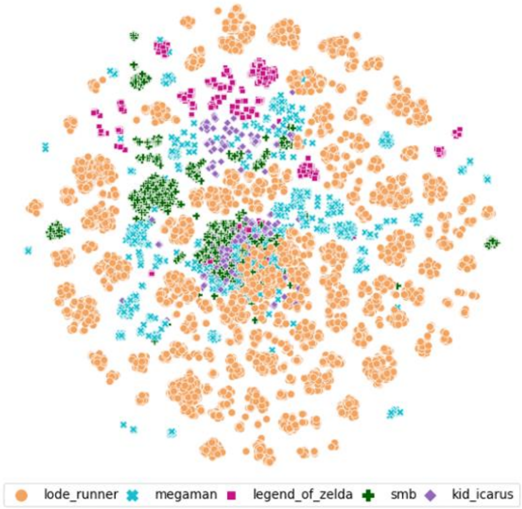
Snodgrass and Ontanon, 2016



Sarkar, Yang & Cooper, 2019



Guzdial and Riedl, 2018

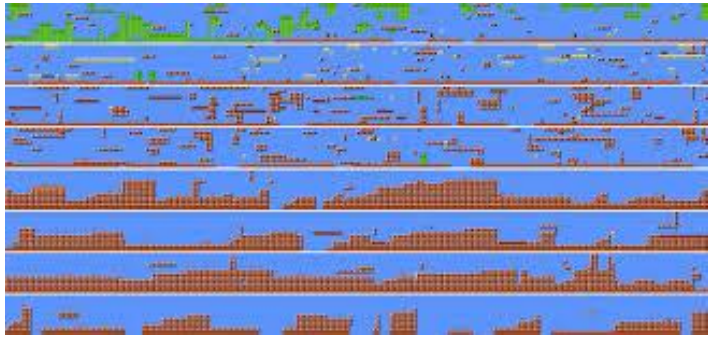


Jadhav and Guzdial, 2021

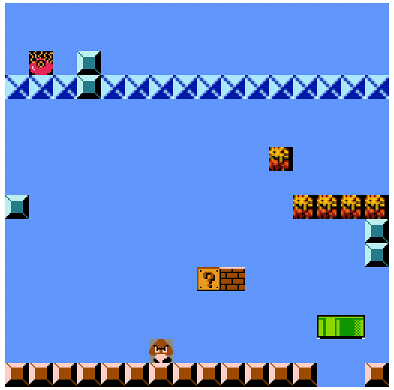
Motivation

- Recent PCGML research has focused on learning and reasoning about design spaces spanning a number of games:
  - Generating new games by recombining learned game graphs
  - Blending latent representations
  - Affordance-based tile embeddings across multiple games
  - Domain adaptation

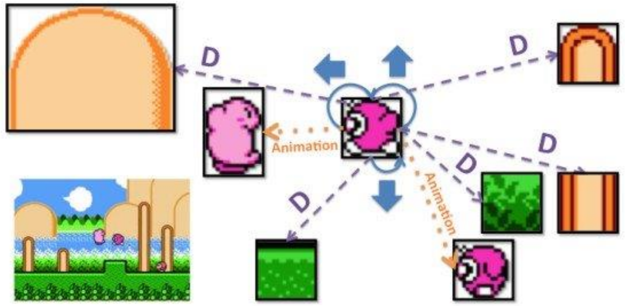
- Potential application: Videogame style transfer



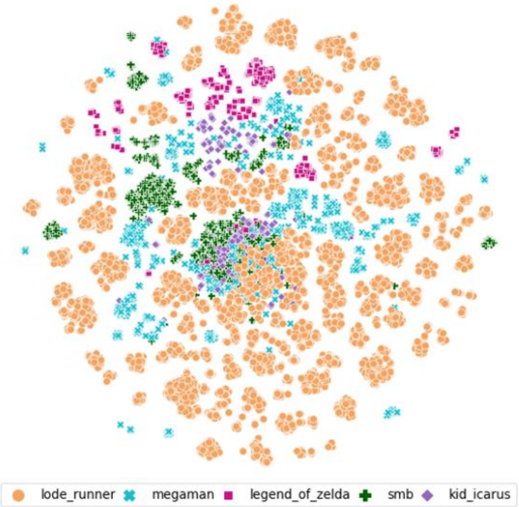
Snodgrass and Ontanon, 2016



Sarkar, Yang & Cooper, 2019



Guzdial and Riedl, 2018



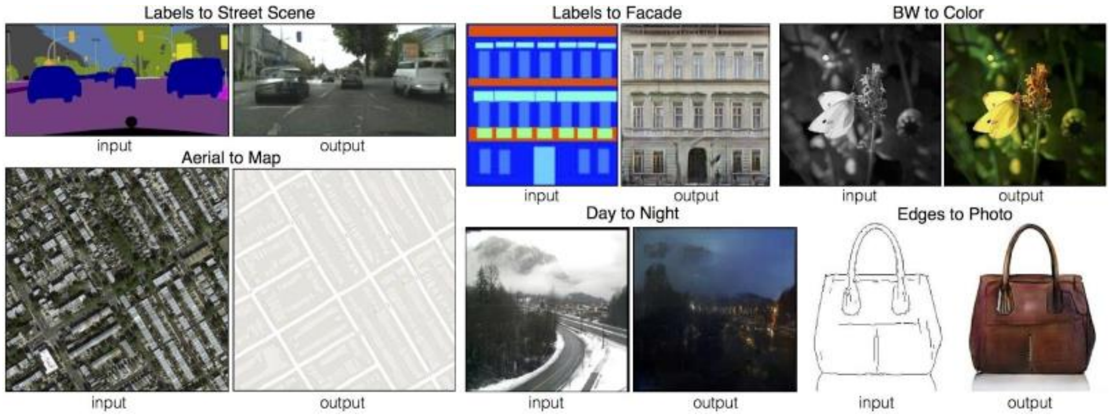
Jadhav and Guzdial, 2021

Motivation

- Style transfer
  - Neural style transfer introduced by Gatys et al. and has become a popular application of ML in visual art

  - pix2pix --- learning translation functions between sets of images by training on pairs of images, allowing for style transfer-adjacent applications between different image domains

Gatys et al., 2015

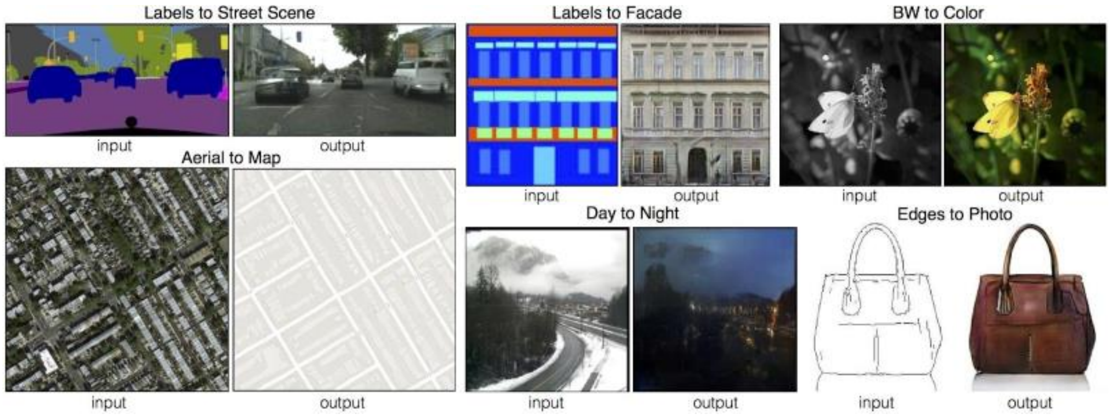


Isola et al., 2017

Motivation

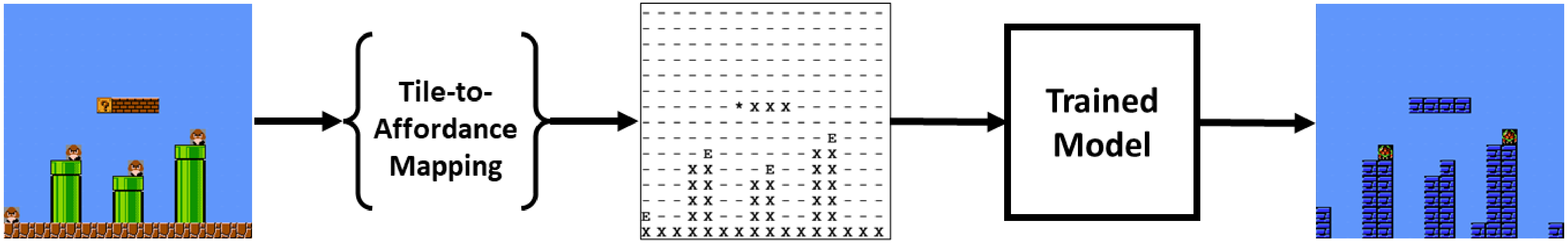
- Style transfer
  - Neural style transfer introduced by Gatys et al. and has become a popular application of ML in visual art
  - pix2pix --- learning translation functions between sets of images by training on pairs of images, allowing for style transfer-adjacent applications between different image domains
  - But style transfer for games remains underexplored
    - No prior works explicitly attempting style transfer for games beyond pixel-based approaches; tile-based approach may be more suitable for considering functional aspect of levels

Gatys et al., 2015



Isola et al., 2017

tile2tile

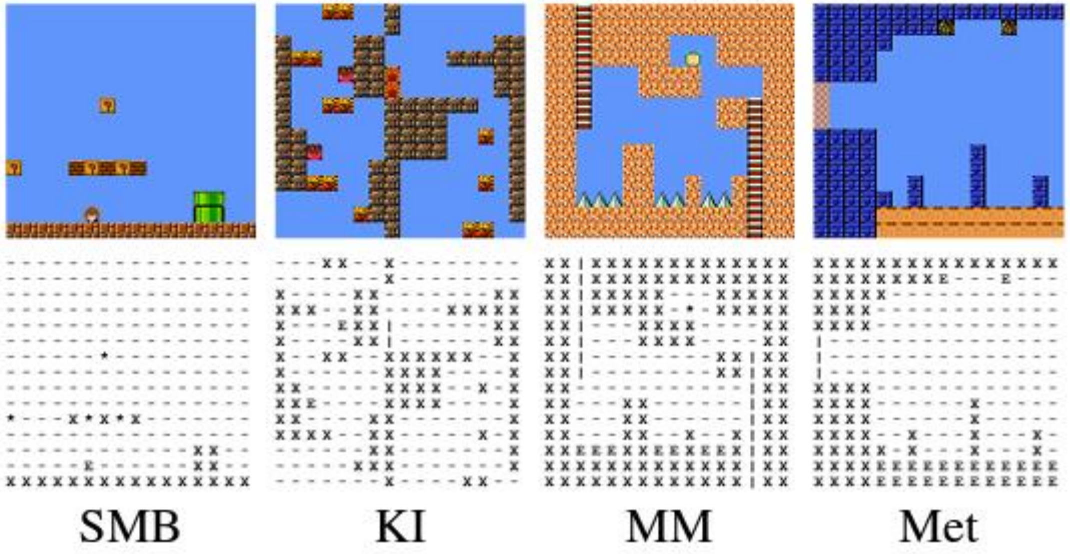


- A framework for transferring style between levels of two games--A (source) and B (target)--by:
  - Converting the level from source game A into an affordance-based representation (sketch)
  - Translating the sketch into the original tiles of target game B using a trained model (filter)

- Similar to pix2pix (Isola et al. 2017)
  - Translate between sets of tile-based levels instead of sets of images

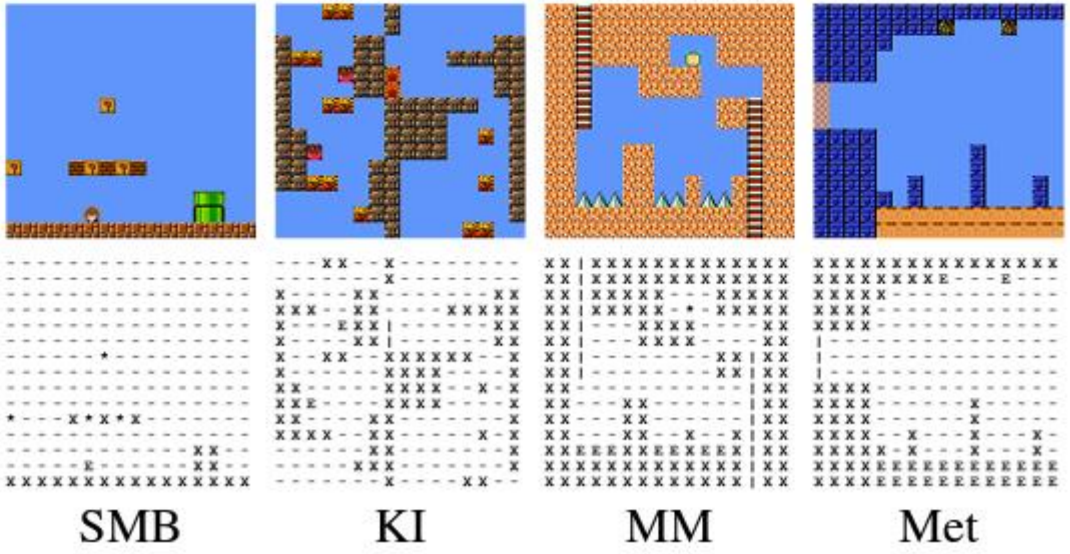
tile2tile

- Affordances of an object refer to the types of in-game actions and interactions that it permits
- Games differ in their specific tile-based representations but can share unified affordance representations
  - e.g. Mario has goombas, Metroid has Metroid creatures, but both are enemies



tile2tile

- Affordances of an object refer to the types of in-game actions and interactions that it permits
- Games differ in their specific tile-based representations but can share unified affordance representations
  - e.g. Mario has goombas, Metroid has Metroid creatures, but both are enemies
- For each game:
  - Defined a mapping from its original tiles to common shared affordances
  - Trained a model (filter) to convert affordance representation to that game's tiles
- Models used:
  - Markov Random Field
  - Convolutional Autoencoder



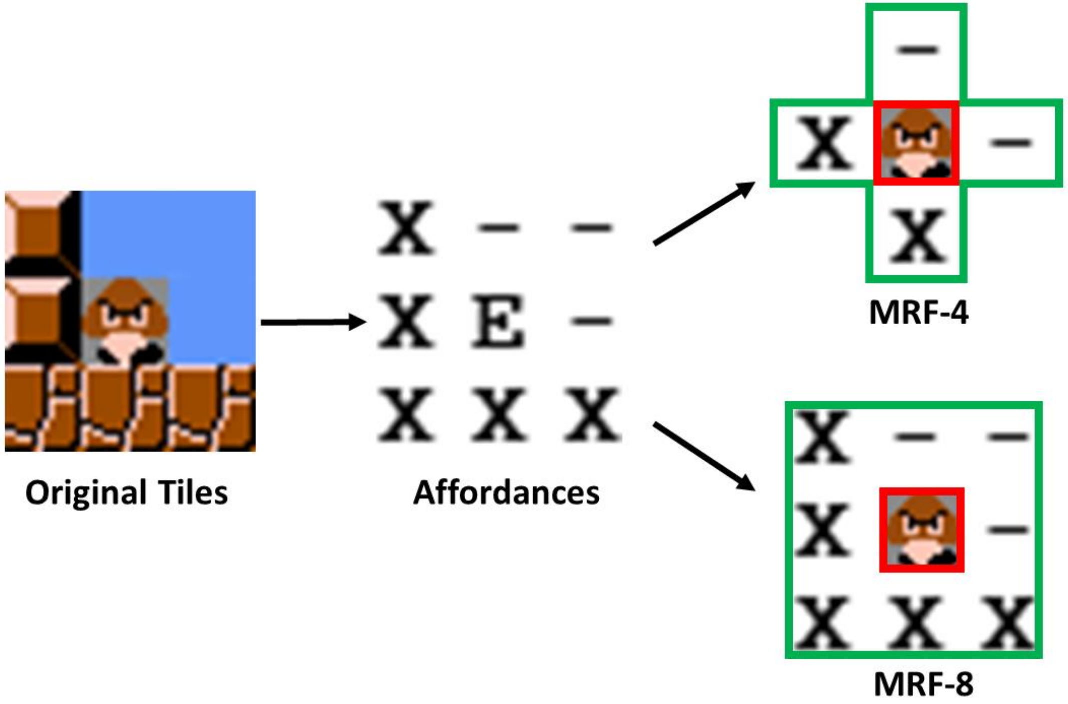
Aff	SMB	KI	MM	Met
X				
E				
|	None			
⋆		None		None

Markov Random Field (MRF)

- MRFs learn conditional probability distributions for any given state conditioned on its surrounding states

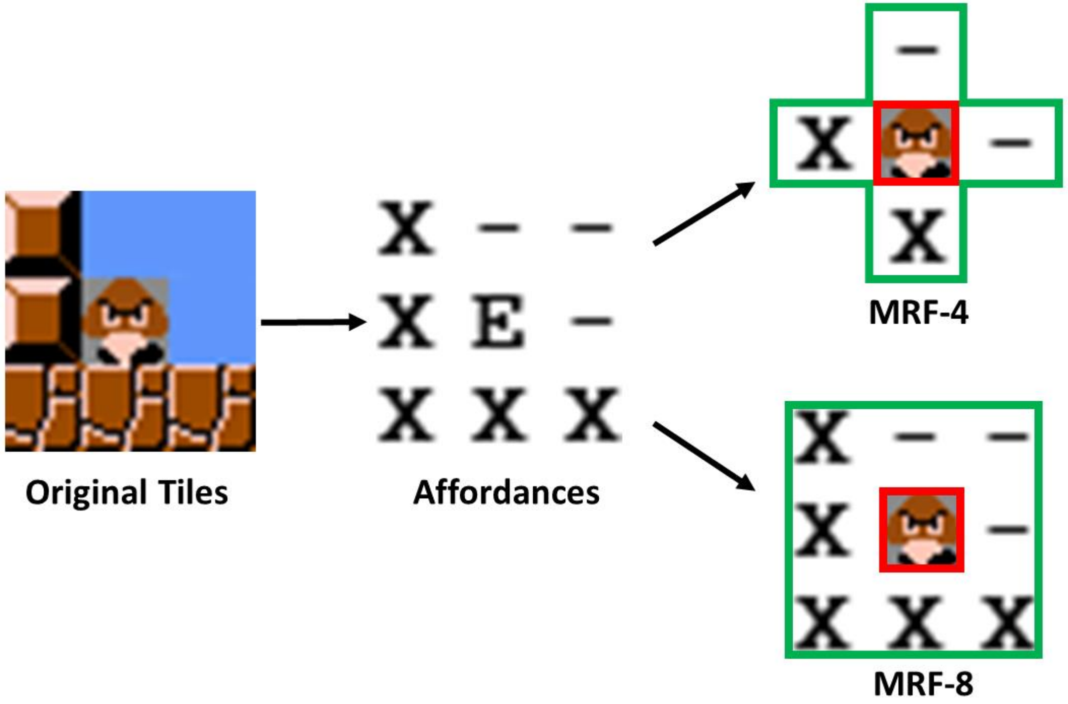
- In tile2tile, role of model is to learn mapping from game-agnostic affordances to game-specific tiles
  --- MRF models this as learning the probability of a game-specific tile at a location given the affordances of the surrounding neighborhood (context) of that location
  --- MRF for a game thus learns the probability of tile occurrences given a surrounding affordance context

- Tested 2 contexts
  - 4 tiles (N, S, E, W)
  - 8 tiles (N, S, E, W, NE, NW, SE, SW)

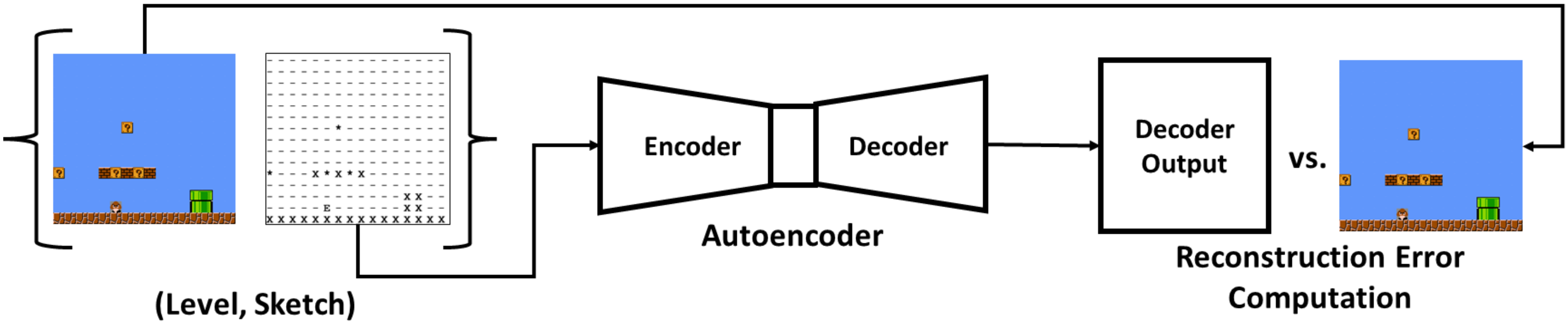


Markov Random Field (MRF)

- MRFs implemented as dictionaries
  - Key: unique affordance context
  - Value: distribution of center tiles for the target game given that surrounding context
- To train MRFs for a given game
  - Convert each level to affordances
  - Slide 3x3 MRF window across each level and update the counts of original center tiles for each unique context
  - Convert each context's tile counts into a distribution that we can sample from



Convolutional Autoencoder (AE)



- For each game, trained a model on 15x16 segments
- Collected pairs of level segments i.e. original and sketch versions of each segment
- Sketch used as encoder input, recon loss calculated between decoder output and original segment
- Training process analogous to denoising autoencoders

Style Transfer Process

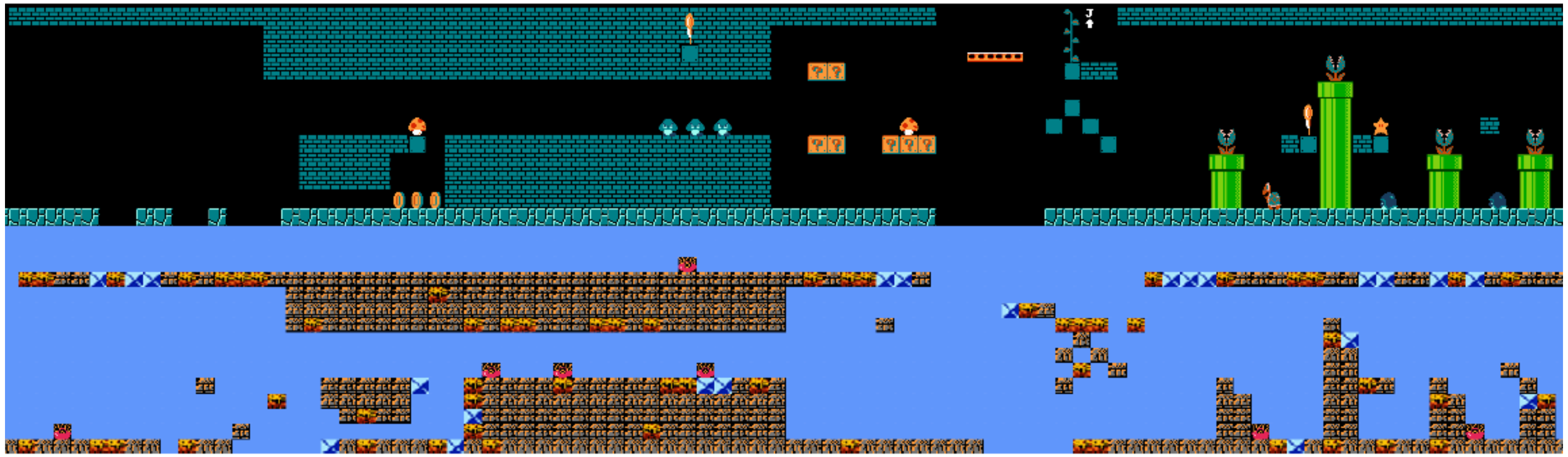
- Two-step process
  - Convert level from source game A to affordance-based representation (sketch) using pre-defined tile-to-affordance mapping
  - Apply target game B's trained model (filter) on converted level sketch:
    - MRF: slide 3x3 trained MRF across source level, converting tile-by-tile
    - AE: forward 15x16 source segment through game B's AE

Style Transfer Process

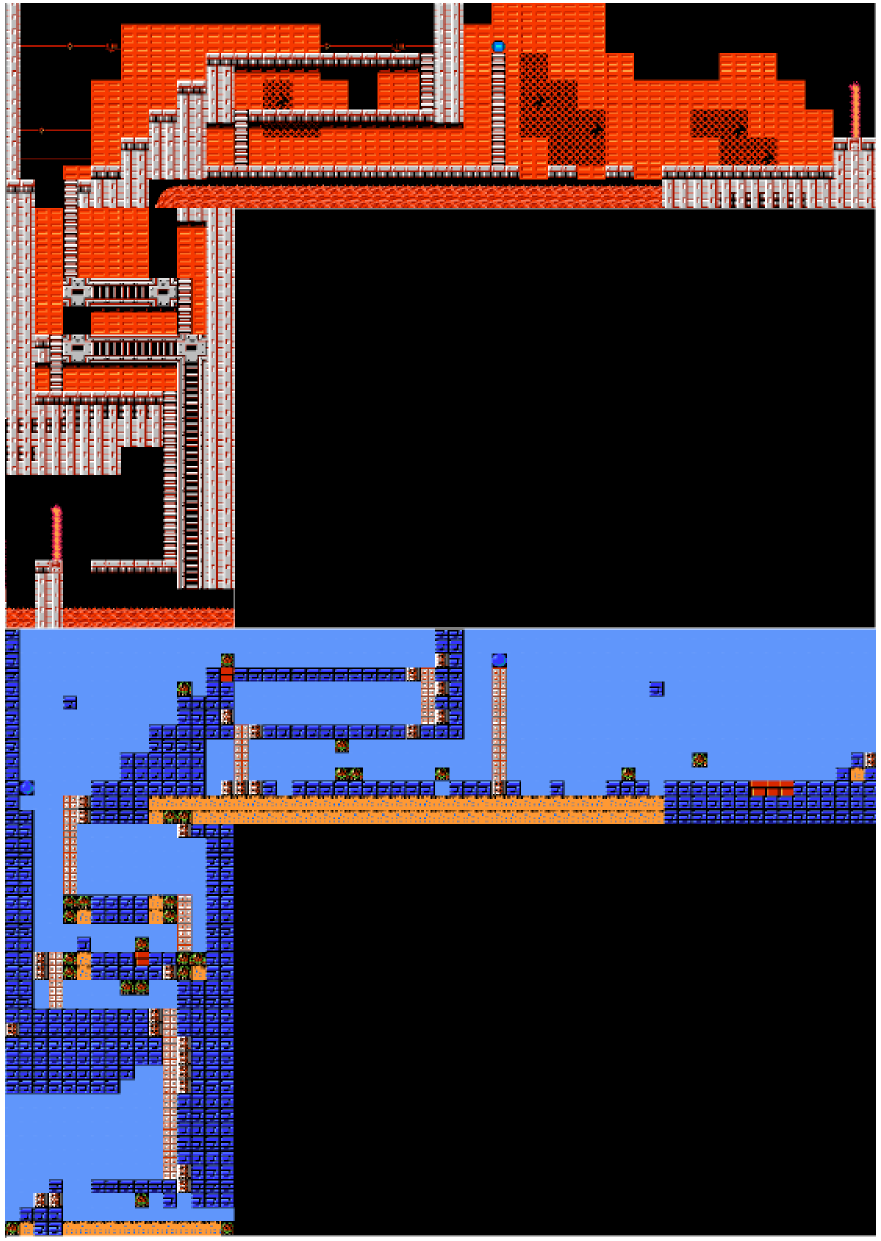
- Two-step process
  - Convert level from source game A to affordance-based representation (sketch) using pre-defined tile-to-affordance mapping
  - Apply target game B's trained model (filter) on converted level sketch:
    - MRF: slide 3x3 trained MRF across source level, converting tile-by-tile
    - AE: forward 15x16 source segment through game B's AE

- Differences
  - MRF by design keeps underlying affordances fixed while translating the overarching game-specific tiles; AE generates underlying content and game-specific tiles from scratch
  - MRF performs style-transfer tile-by-tile; AE performs style-transfer segment-by-segment

MRF Examples

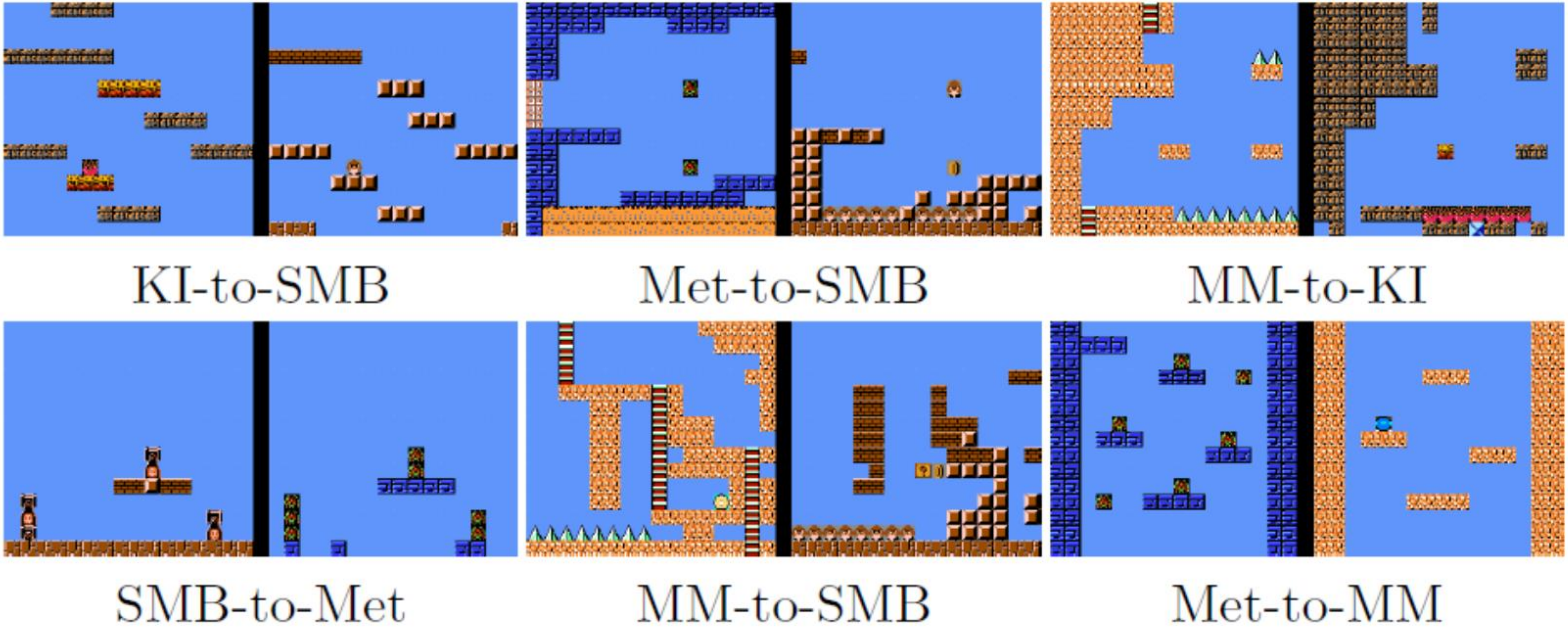


Mario (top) to Kid Icarus (bottom)



Mega Man (top) to Metroid (bottom)

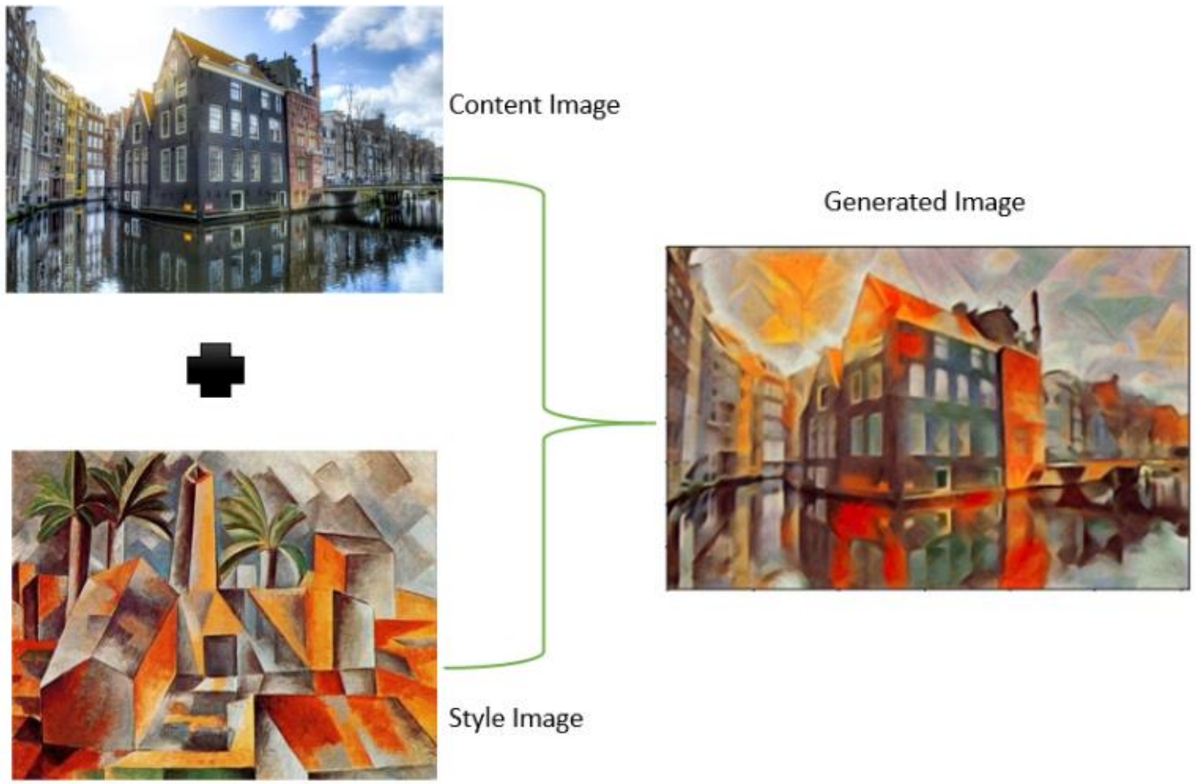
AE Examples



Content/Style and Affordances/Tiles

- Affordances and Tiles analogous to style transfer concepts of Content and Style
  - Affordances/Content represent underlying level topology/image content
  - Tiles/Style represent high-level game-specific tiles/high-level visual characteristics

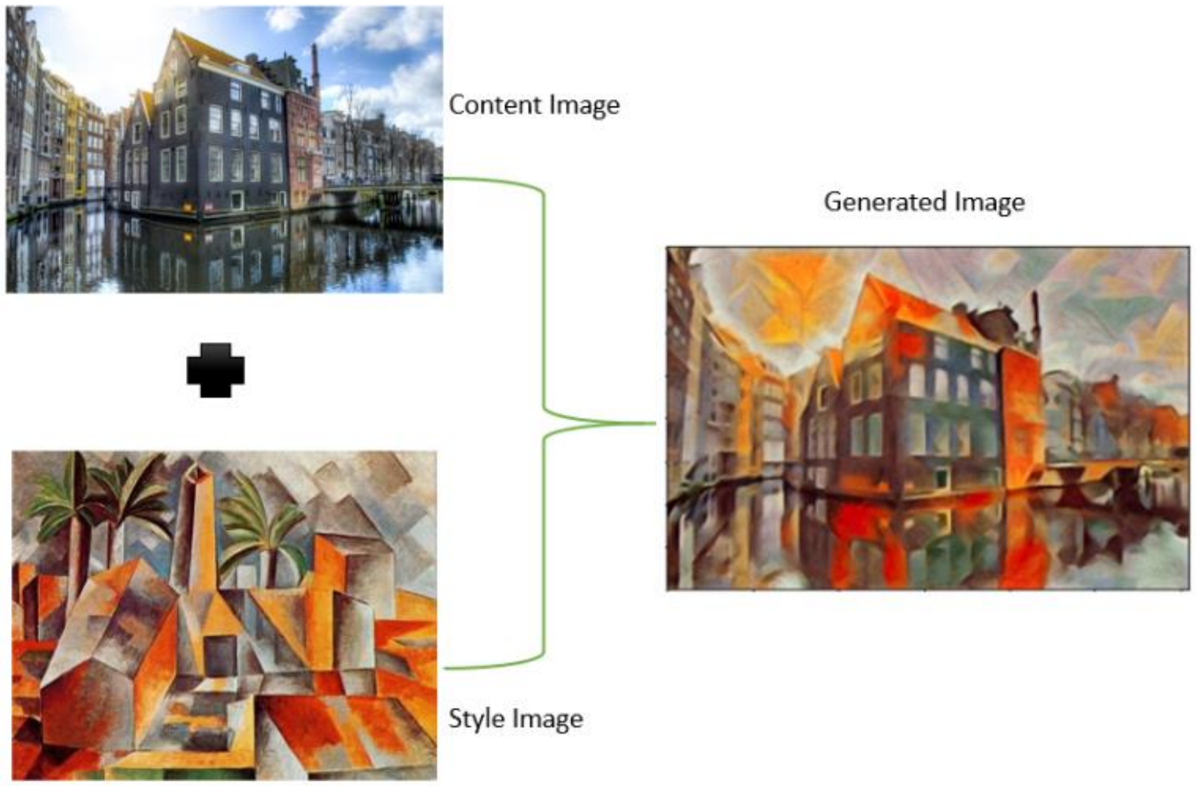
- Goal: keep underlying affordances similar while translating high-level game-specific tiles



Source: https://www.pluralsight.com/guides/artistic-neural-style-transfer-with-pytorch

Content/Style and Affordances/Tiles

- Affordances and Tiles analogous to style transfer concepts of Content and Style
  - Affordances/Content represent underlying level topology/image content
  - Tiles/Style represent high-level game-specific tiles/high-level visual characteristics
- Goal: keep underlying affordances similar while translating high-level game-specific tiles
- Three-part evaluation
  - Content
  - Style
  - Playability



Source: https://www.pluralsight.com/guides/artistic-neural-style-transfer-with-pytorch

Content Evaluation

- Use Affordance Pattern KL-Divergence (APKLDiv) metric to compare tile affordance patterns in input source and output target levels
  - Same as Tile Pattern KL-Divergence metric (Lucas and Volz) but use the affordance tiles rather than original game tiles
  - APKLDiv can be viewed as analogous to the content loss from visual art style transfer.

Content Evaluation

- Use Affordance Pattern KL-Divergence (APKLDiv) metric to compare tile affordance patterns in input source and output target levels
  - Same as Tile Pattern KL-Divergence metric (Lucas and Volz) but use the affordance tiles rather than original game tiles
  - APKLDiv can be viewed as analogous to the content loss from visual art style transfer.

- For MRFs, since the style transfer process by design keeps the affordances fixed from source to target, this evaluation is not useful; however, useful evaluation for autoencoders since they have to learn to maintain affordance similarity

Content Evaluation

- For every pair of games S (source) and T (target)
  - OG-Source: original segments of source S
  - TF-Target: segments obtained by applying autoencoder of target game T
  - Compute the APKLDiv between these 2 sets using their sketch (affordance) forms

  - OG-Target: original segments of target T
  - Also compute APKLDiv between TF-Target and OG-Target (both in sketch form)
  - For effective style transfer, expect:

    APKLDiv(TF-Target, OG-Source) < APKLDiv(TF-Target, OG-Target)

- E.g. SMB-to-KI style transfer (Source S: SMB, Target T: KI)
  - The underlying affordances (content) of generated, style-transferred KI levels should be closer to SMB-like affordances than KI-like affordances

Source-Target	TF-Target vs OG-Source	TF-Target vs OG-Target
KI-SMB	$\mathbf{0.71 \pm 0.61}$	1.52 ± 1.36
MM-SMB	$\mathbf{1.39 \pm 1.15}$	2.13 ± 1.70
Met-SMB	$\mathbf{1.32 \pm 1.19}$	2.05 ± 1.63
SMB-KI	$\mathbf{0.27 \pm 0.22}$	0.87 ± 0.58
MM-KI	$\mathbf{1.34 \pm 1.11}$	1.99 ± 1.6
Met-KI	$\mathbf{1.17 \pm 1.02}$	1.72 ± 1.38
SMB-MM	$\mathbf{0.45 \pm 0.32}$	1.08 ± 0.62
KI-MM	$\mathbf{0.38 \pm 0.36}$	1.8 ± 0.56
Met-MM	$\mathbf{0.82 \pm 0.61}$	1.23 ± 0.98
SMB-Met	$\mathbf{0.19 \pm 0.16}$	1.22 ± 0.74
KI-Met	$\mathbf{0.38 \pm 0.38}$	2.31 ± 2.14
MM-Met	$\mathbf{0.97 \pm 0.74}$	1.67 ± 1.45

Style Evaluation

- Game-specific style is defined by the game-specific tile representation (i.e. tile histogram) of underlying affordance patterns

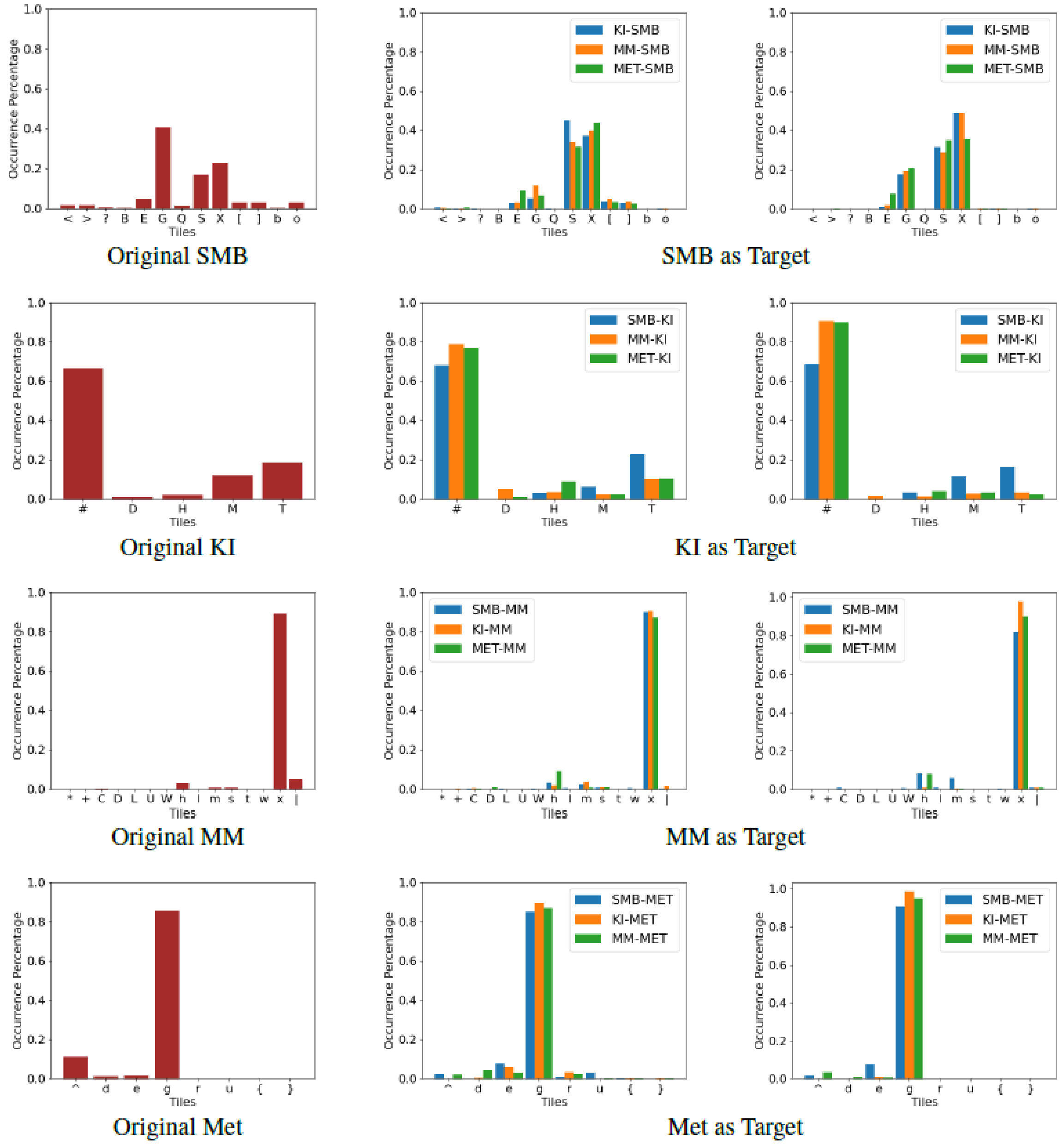
- For each pair of games source S and target T
  - Applied the model (MRF/AE) for T on each level of S and computed the tile histogram of the style-transferred levels
  - Compared this tile histogram with original tile histogram of target game T

Style Evaluation

- Game-specific style is defined by the game-specific tile representation (i.e. tile histogram) of underlying affordance patterns

- For each pair of games source S and target T
    - Applied the model (MRF/AE) for T on each level of S and computed the tile histogram of the style-transferred levels
    - Compared this tile histogram with original tile histogram of target game T

- Tile distributions obtained after style transfer are similar to those of the original levels suggesting that style-transferred levels have similar tile distributions as the original target levels



Playability Evaluation

- Evaluated both models using 15x16 segments on which we ran game-specific tile-based A* agents (from VGLC) where each game's agent can perform game-specific jumps
- For each source-target game pair, computed the percentage of segments that were playable after style transfer using the target game's model
- Takeaways
  - Performance varies for different game pairs
  - Obtain good playability percentages for several pairs
  - Reasonable that transferring style would lead to a dip in playability

	MRF-4	MRF-8	AE-128	AE-256
KI-SMB	75	67.5	**81.25**	75
MM-SMB	39.86	43.36	**80.42**	77.62
Met-SMB	33.07	31.65	72.87	**79.54**
SMB-KI	**75.57**	71.59	62.5	60.8
MM-KI	44.76	46.85	**49.65**	48.25
Met-KI	**48.79**	46.37	32.87	39.31
SMB-MM	**71.34**	68.18	59.09	64.21
KI-MM	**69.23**	63.75	60	67.5
Met-MM	32.63	33.47	38.39	**48.28**
SMB-Met	**85.8**	81.25	60.23	63.07
KI-Met	72.15	**73.75**	61.25	65
MM-Met	46.77	52.45	**61.54**	60.14

Future Work

- Validate style transfer using user studies and develop an application for users to apply trained filters on pre-existing or hand-authored levels

- Style-transferred levels could be used to supplement levels for games with insufficient training data

- Generalize to games of other genres

Future Work

- Validate style transfer using user studies and develop an application for users to apply trained filters on pre-existing or hand-authored levels
- Style-transferred levels could be used to supplement levels for games with insufficient training data
- Generalize to games of other genres

Contact

Anurag Sarkar

Northeastern University

sarkar.an@northeastern.edu