Runway: Adding artificial intelligence capabilities to design and creative platforms

Cristóbal Valenzuela
New York University, Runway
cristobal@runwayapp.ai

Alejandro Matamala
New York University, Runway
alejandro@runwayapp.ai

Anastasis Germanidis
Runway
anastasis@runwayapp.ai

Abstract

Considering that the research and development of artificial intelligence systems will continue to grow over time, the inherent complexity and segmentation of those systems will expand as well. Given this scenario, new ways need to be created to open up its potential to practitioners coming from other disciplines. We present Runway, an interface and framework that orchestrates the training, use and deployment of artificial intelligence models in design and creative platforms. Runway facilitates human-AI collaboration using a language that is familiar to creators from a broad range of disciplines.

1 Introduction

Creative applications of machine learning have grown significantly over the last few years [1] [2]. Generative explorations of neural network based models have created a new wave of AI related works, conferences, workshops and discussions. Yet, one of the main challenges in the creation of creative and artistic work with machine intelligence has been the domain expertise needed to create a usable ML pipeline. Training a model or using it for inference requires specific expertise in the subject. This means that only artists, designers and creators with the necessary technical background are able to explore the creative capabilities of AI. A common industry approach towards this challenge has been to update and expand current design software to include custom built in AI competences [3]. Upgrading existing platforms to enable AI functionality requires platform specific tool chains, environments, languages and it's limited to the environment capabilities on which the software is built. The presence of a wide range of frameworks [4] [5] used to create machine-learning models also imposes a restriction in terms of interoperability with this software.

Rather than developing platform-specific AI functionalities, this work attempts to create a new toolkit that operates as an in-between layer between creators and design platforms, making AI models accessible, discoverable and usable by creative practitioners of different fields.

2 Machine Learning Orchestration

Runway is an open ended toolkit for exploring the possibilities of human-AI collaboration. It is a GUI, model ecosystem, and set of integrations, enabling the use of pre-trained models inside Docker-containerized environments running locally or in a remote GPU provider. Users can browse the Runway model directory to discover and install new models, and then experiment with those models inside Runway or integrate them with design and creative platforms (including Adobe Illustrator, Ableton, Processing, and Unity) through network protocols such as HTTP, WebSockets, and OSC.

Runway is logically organized in a hierarchy of Workspaces and Models. A Runway Model consists of a Docker image that serves an ML model, as well as associated metadata such as description, usage details, and links to related resources (e.g. academic papers, Github repositories). Workspaces provide a way to logically group models used in the same project together. For example, a user

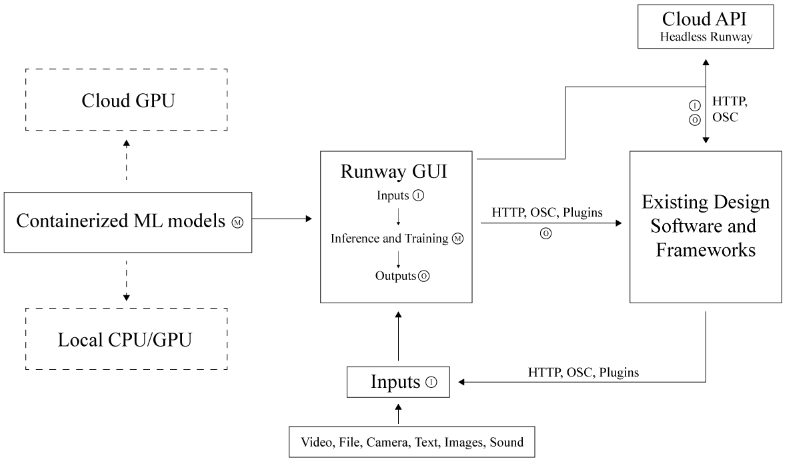
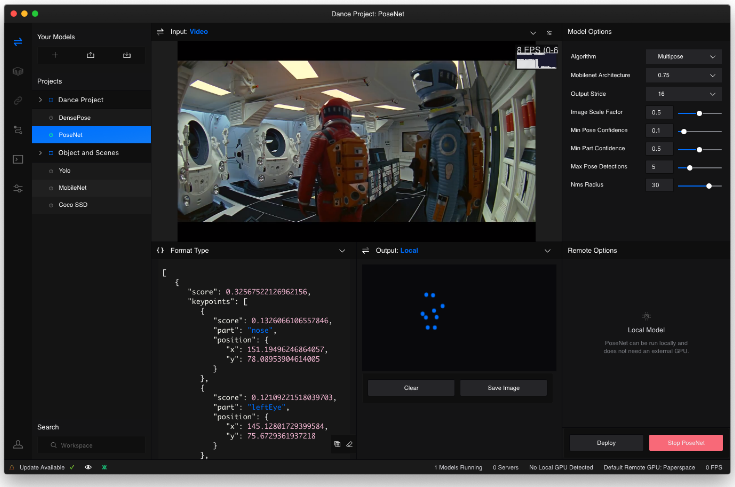


Figure 1: The Runway architecture



Algorithm	Multipose
Mobilenet Architecture	0.75
Output Stride	16
Image Scale Factor	0.5
Min Pose Confidence	0.1
Min Part Confidence	0.5
Max Pose Detections	5
Nms Radius	30

```
[
  {
    "score": 0.32567522126962156,
    "keypoints": [
      {
        "score": 0.1326066106557846,
        "part": "nose",
        "position": {
          "x": 151.19496246864057,
          "y": 78.08953904614005
        }
      },
      {
        "score": 0.12109221518039703,
        "part": "leftEye",
        "position": {
          "x": 145.12801729399584,
          "y": 75.6729361937218
        }
      },
```

PoseNet can be run locally and does not need an external GPU.

Update Available 1 Models Running 0 Servers No Local GPU Detected Default Remote GPU: Paperspace 0 FPS

Figure 2: Runway I/O view

working an interactive dance performance may create a workspace containing two models that she would like to use for her piece: OpenPose [6] for detecting the dancers' movements and WaveNet [7] for generating a score for the dance.

3 Conclusion

Runway started with the idea that a much greater number of artists, designers, and other creators would be interested in using machine learning models to augment their workflow if only they had the right tools built for them. By introducing a specification for running containerized machine learning models, a simple visual interface for experimenting with those models, and a framework for integrating them in existing design software, this work attempts to shift the energy of creators from installing platform-specific dependencies to creating and experimenting.

References

[1] L. Gatys, A. Ecker, and M. Bethge. A neural algorithm of artistic style. Nature Communications, 2015.

[2] Elgammal, A., Liu, B., Elhoseiny, M., Mazzone, M.: CAN: Creative adversarial networks, generating art by learning about styles and deviating from style norms. arXiv preprint arXiv:1706.07068, 2017.

[3] Adobe. Adobe Sensei. https://www.adobe.com/sensei.html, 2018.

[4] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Mike Schuster, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[5] Paszke, Adam and Gross, Sam and Chintala, Soumith and Chanan, Gregory and Yang, Edward and DeVito, Zachary and Lin, Zeming and Desmaison, Alban and Antiga, Luca and Lerer, Adam. Automatic differentiation in PyTorch. NIPS-W, 2017.

[6] Z. Cao, T. Simon, S. Wei, and Y. Sheikh. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. CVPR, 2017.

[7] S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, “Wavenet: A generative model for raw audio,” arXiv:1609.03499, 2016.