ImJoy: A computational platform for the deep learning era

Wei Ouyang
Cell Profiling Group, SciLifeLab, KTH, Stockholm
Headed by Emma Lundberg
Overview

- Human Protein Atlas and HPA image classification competition
- ANNA-PALM: Deep learning for super-resolution microscopy
- ImJoy: A computational platform for the deep learning era
The Human Protein Atlas

Mapping the human proteome using in-house generated proteome-wide collection of antibodies.
Using large-scale immunostaining and high-resolution microscopy
Freely available database: www.proteinatlas.org

HPA Cell Atlas

Prof. Lundberg
“Foreign Cell Sample” Classification

Uhlen et al, A tissue based map of the human proteome, Science, 2015
Thul et al, A subcellular map of the human proteome, Science, 2017
Sullivan et al, D, Nature Biotechnol. 2018

>322,000 players on EVE online
32,000,000 classifications (70 working years)
Featured Prediction Competition

Human Protein Atlas Image Classification
Classify subcellular protein patterns in human cells

Human Protein Atlas · 2,172 teams · 4 months ago

- >120,000 annotated images
- $37,000 prize money
- 3 months
- >2,000 teams
- >55,000 submissions
- Top model: Densnet 121 + lovaza loss etc.
- 20% higher score than previous record

Deep learning + large image dataset + crowdsourcing!

Ouyang et. al, Nat Methods, 2019
Deep learning accelerated
Super-resolution microscopy
Artificial Neural Network Accelerated PALM/STORM

raw images → 300 frames → 30,000 frames

Super-res Reconstruction Error
L1 + MS-SSIM

Low-res Reconstruction Error
Ak PK MS-SSIM

Conditional GAN Error
L Pk PK Ak MSE

ANNA-PALM
Artificial Neural Network Accelerated PALM

Betzig et al. Science 2006  Ouyang et. al, Nat Biotechnol, 2018
ANNA-PALM: 100x faster SR imaging

Input
PALM (k=300)

Output
ANNA-PALM (k=300)

Ground-truth
PALM (K=30,000)

Ouyang et. al, Nat Biotechnol, 2018
Web Application for ANNA-PALM

- No installation required
- Not Scalable
- Data transmission
- Maintenance required!

- Cost
- Privacy, GDPR

Useful, but requires improvements!
ImJoy: An open-source computational platform for the deep learning era
Software for the deep learning area

- Large dataset, remote storage, online database (IDR, HPA)
- Computationally hungry (GPU/TPU/NPU...)
- Interactively inspect, annotate data and train model (human-in-the-loop)

Inspection with Class Activation Maps

Ouyang et. al, Nat Methods, 2019
Opportunities

- **Deep learning libraries**: Keras/Tensorflow, Pytorch
- **Mobile** is the biggest platform
  - > 5 billion mobile devices
- **Cloud** computing and remote data access
  - cloud services: storage, GPU, database, serverless, AutoML, Kubernetes
  - Free computation: Google Colab, [MyBinder.org](http://MyBinder.org)
  - Remote Storage: S3, N5, Zarr, Dask Array
- **Web** standards
  - HTML5 standards, JS libraries, nodejs/npm
  - UI standards (e.g. material design)
  - Progressive Web App, offline support
  - **WebAssembly** (porting C/C++, C#, Rust, Python)
  - **WebGPU** (deep learning in the browser)
- Hybrid computing modes
  (browser + local + cloud)

- Progressive Web App ⇒ Offline support, mobile support

- Web Assembly/Web GPU ⇒ browser based computing, security

https://imjoy.io
Minimal but powerful!
Key ideas

- Each plugin import and export a set of service functions

- Transparent, symmetrical Remote Procedure Calls (RPC) ⇒ across plugins/programming language/host (also see RPyC in Python)

- Asynchronous execution ⇒ dynamic workflow composition

(Inspired by Crossbar.io, Jailed.js, Tensorflow, Pytorch)
import asyncio

from imjoy import api

class ImJoyPlugin():
    async def setup(self):
        self.pluginA = await api.getPlugin('pasteur.fr/pluginA')
        self.pluginB = await api.getPlugin('kth.se/pluginB')
        self.pluginC = await api.getPlugin('pluginC')

    async def concurrent_workflow(self, x):
        promiseA = self.pluginA.process(x)
        promiseB = self.pluginB.process(x)
        resultA, resultB = await asyncio.gather(promiseA, promiseB)
        result = await self.pluginC.process(resultA, resultB)
        return result
**static vs dynamic workflow**

**Fixed node/connection, file based io**
- Single workflow engine

**Define** → **Compile** → **Run**

- Tensorflow 1.x
- Hard to debug!

**Self-organizing, dynamic node**
- Dynamic connection, data streaming

**Engine 1**
- **GUI**
- **Data**

**Engine 2**
- **A**
- **B**
- **C**

**Pytorch**
Static web app: high scalability & availability, ~ zero cost
**ImJoy**

- Plugin repository
  - demo plugins
    - [Image Annotator](#) (annotation)
    - [HPA-UMAP](#) (visualization)
    - [Skin-Lesion-Analyzer](#) (classification)
    - [HPA-Classification](#) (classification)
    - [Interactive Plot](#) (basic example)
    - [DeepBindScan](#) (Genomics)
    - [Noise2Self](#) (Denoising)
    - [ANNA-PALM](#) (super-resolution)
    - [CARE](#) (3D image denoising)
    - [DPNUnet](#) (Segmentation)
    - [ImageJ Demo](#) (PyImageJ)

(• requires plugin engine)
Conclusion

- HPA competition: large image database + deep learning
- ANNA-PALM: Deep learning accelerated SR imaging
- ImJoy makes it easier to deploy deep learning models
  - Progressive Web App, Web Assembly, HTML5
  - Extendable Plugin Engines
  - Transparent Remote Procedure Calls
  - Async workflow composition
  - Static, scalable, high availability, almost zero cost

ImJoy is now published in Nature Methods: https://rdcu.be/bYbGO
(Deep Learning focused Issue)

Work in progress: data viewers, plugin engine, useful plugins!
Acknowledgements

- Thank all the Kaggle participants for their contributions.

Lundberg Lab
- Ulrika Axelsson
- Anna Bäckström
- Jenny Fall
- Christian Gnann
- Martin Hjemare
- Diana Mahdessian
- Anna Martinez Casals
- Rutger Schutten
- Charlotte Stadler
- Devin Sullivan
- Peter Thul
- Casper Winsnes
- Hao Xu
- Lovisa Åkesson
- Anthony Cesnik

Human Protein Atlas
- Mathias Uhlen
- Fredrik Pontén
- Adil Mardinoglu
- Kalle von Feilitzen
- Jan Mulder
- Cecilia Lindskog
- Burcu Ayoglu

Collaborators
- Manu Leonetti, CZ Biohub
- Kathryn Lilley, Cambridge Univ
- Oana Carja, CMU
- Attila Szantner, MMOS
- Andie Nordgren, CCP games
- Bergur Finnbogasson, CCP games
- EVE Online players
Acknowledgements

**Imaging & Modeling Unit @ Institut Pasteur**

- Christophe Zimmer
  - Andrey Aristov, Benoît Lelandais, Christian Weber, Elena Rensen, **Florian Muller**, Jyotsana Parmar
  - Mickael Lelek, Maxime Woringer, Xian Hao)
- Cécile Leduc, Sandrine Etienne-Manneville
- Sandrine Lévêque-Fort, Nicolas Bourg
- Orestis Faklaris, Julien Sellès

**The IT service of Pasteur, Inception GPU lab**

(Stéphane Fournier, Thomas Menard, Jean-Baptiste Denis, Tru Huynh, et. al.)
Thank you!
Useful links for ImJoy

Source code and documentations

- [documentation, API functions](#)
- [main repo, plugin-engine](#)

Example Plugins

- [official plugin repository, example plugins](#)

Getting Help

- [Image.sc Forum](#)
- [ImJoy Slack](#)

Bug Report

- [ImJoy Issues](#)

Preprint on Nature Methods: [https://rdcu.be/bYbGO](https://rdcu.be/bYbGO)