

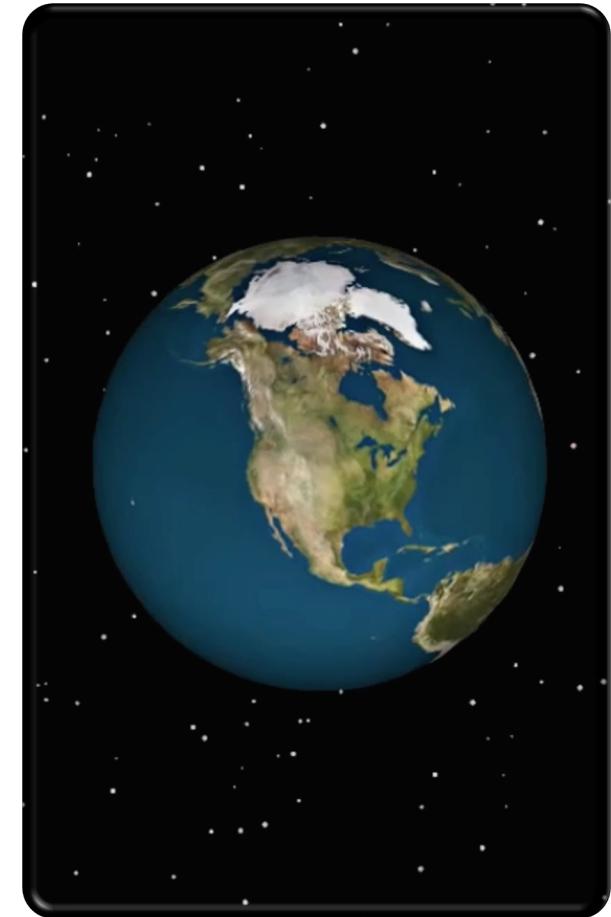
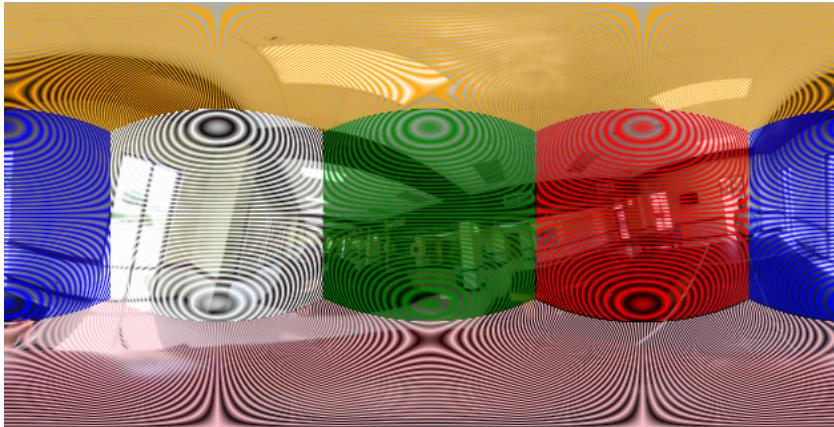
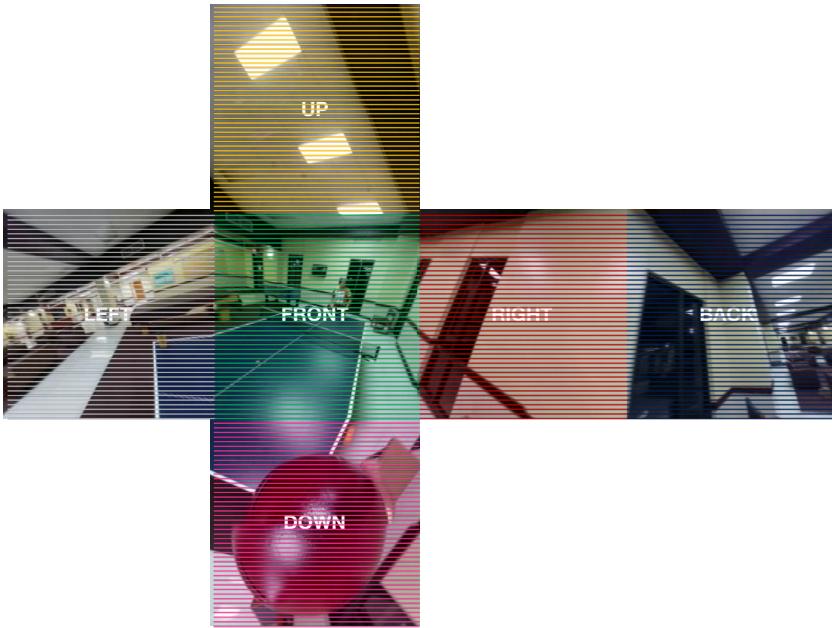
# Parallel - 360 Projector

A portable transformer layer to learn/transfer perspective representation  
to spherical representation.

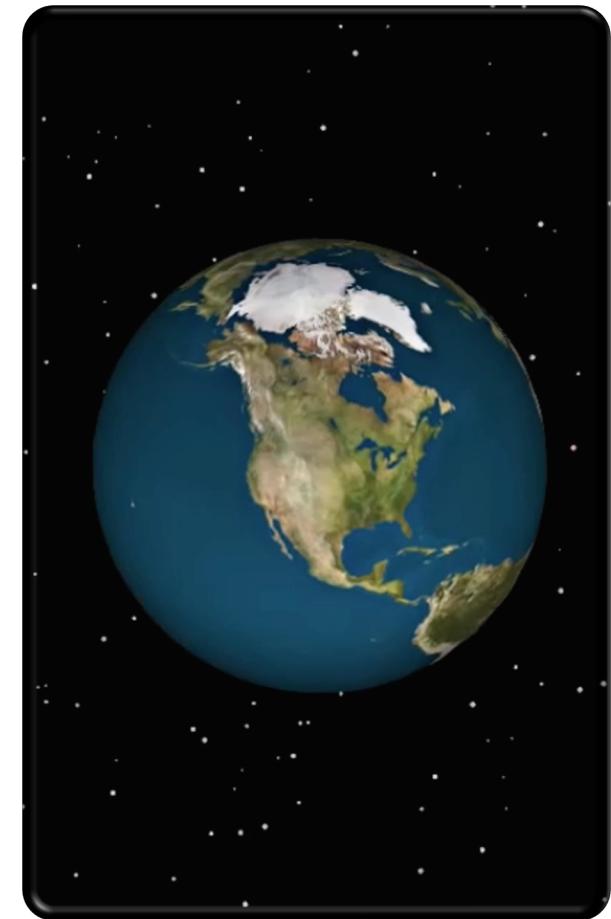
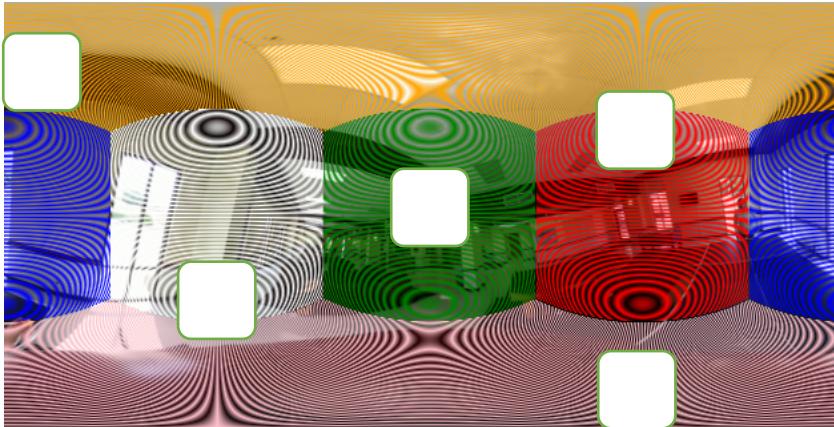
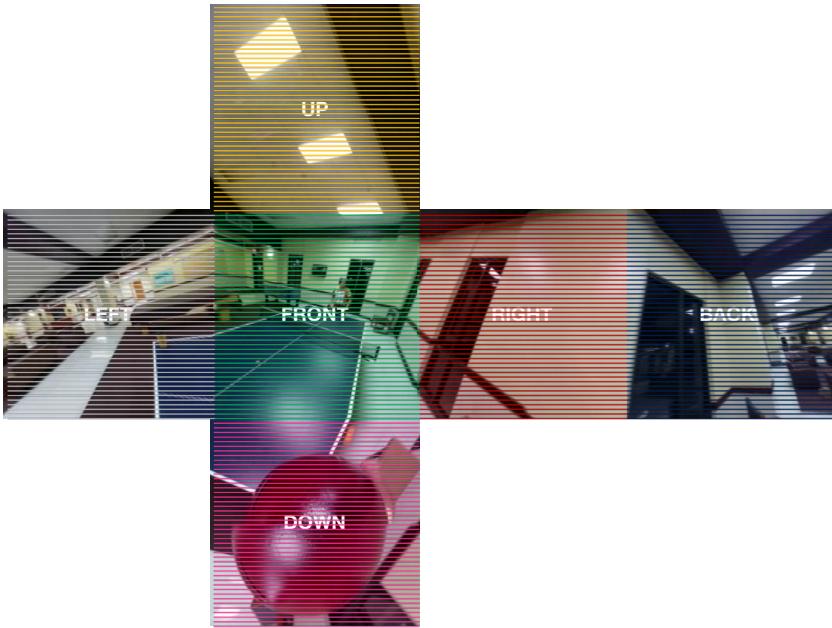
By

Keshav Bhandari

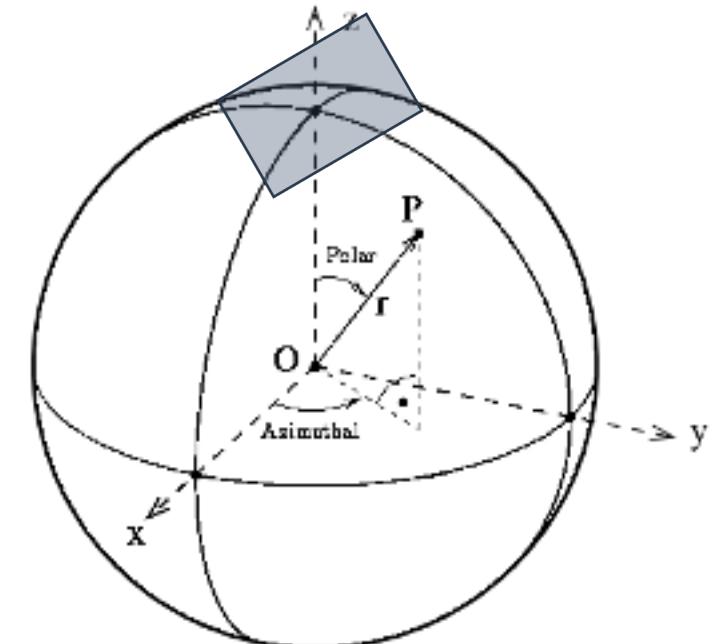
# Motivation



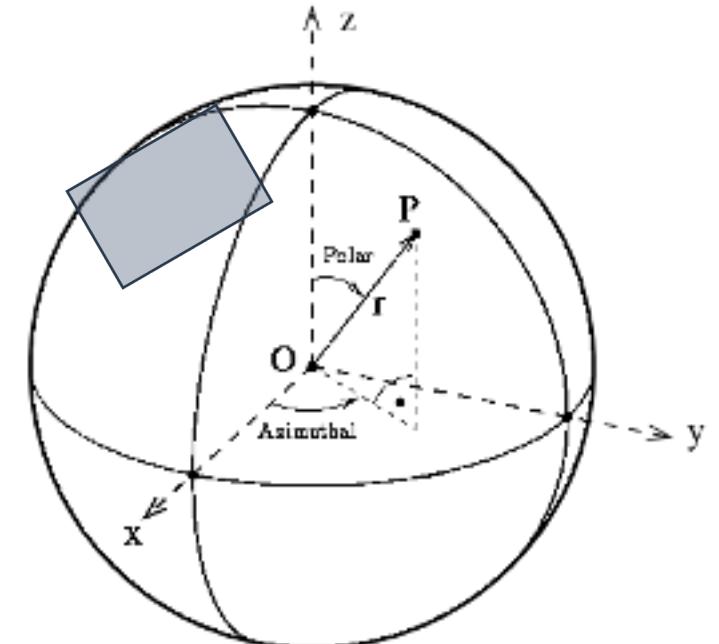
# Motivation



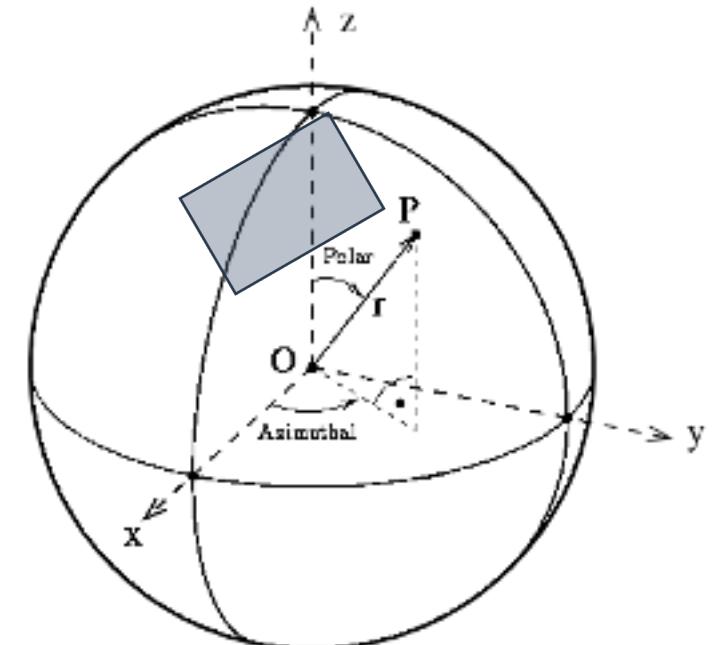
# Motivation



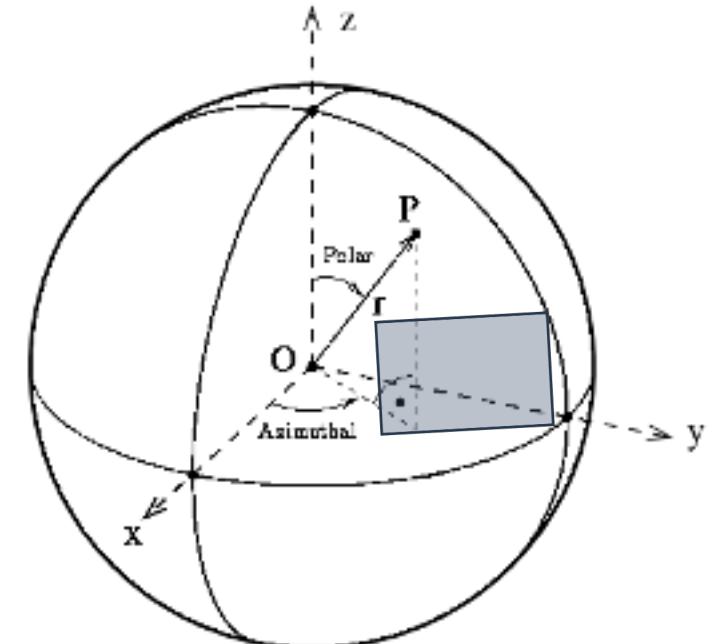
# Motivation



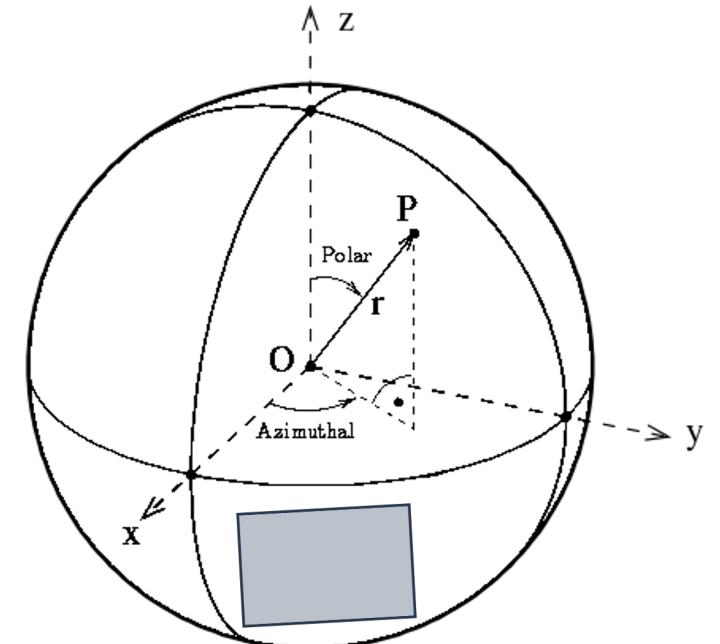
# Motivation



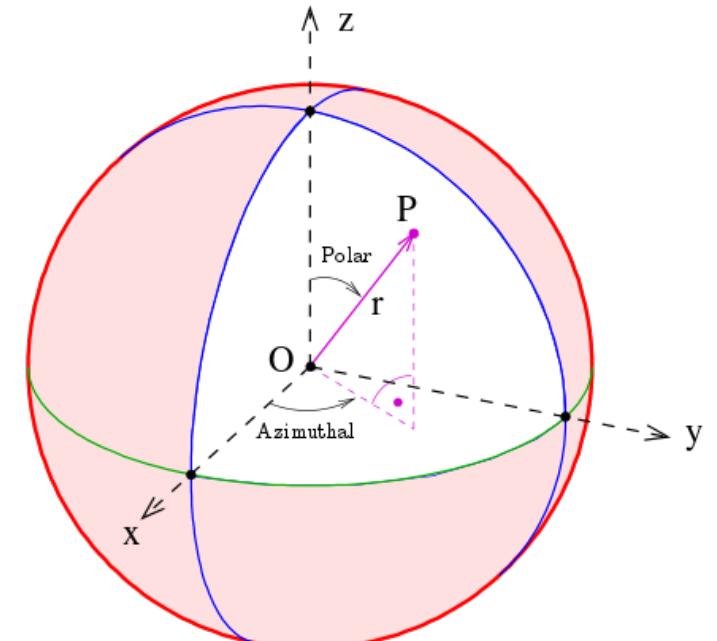
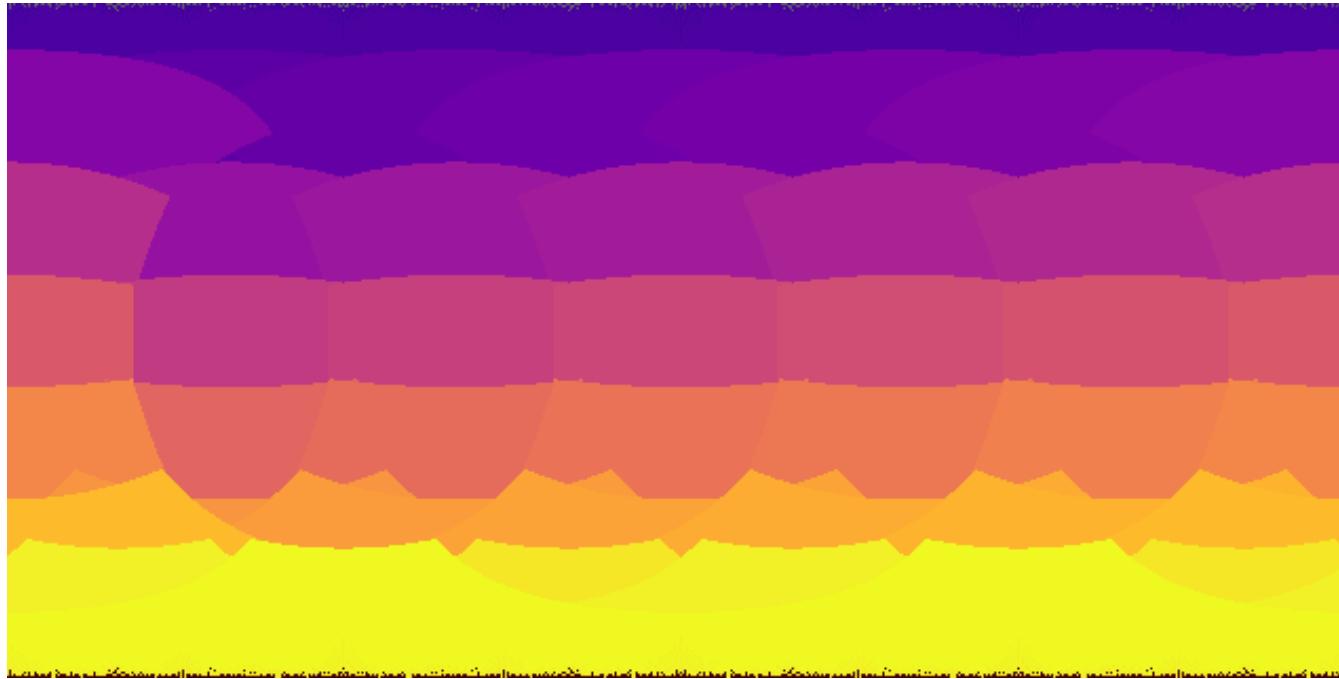
# Motivation



# Motivation



# Motivation



# Motivation

- How to exploit off-the shelf models?
- Are Convolutions/Pooling/Interpolation and other layers okay for 360?

# Math

$$\begin{aligned}x &= \rho \sin\phi \cos\theta \\y &= \rho \sin\phi \sin\theta \\z &= \rho \cos\phi\end{aligned}$$

Spherical to Rectangular

$$\begin{aligned}\rho^2 &= x^2 + y^2 + z^2 \\ \theta &= \arctan(y/x) \\ \phi &= \arccos(z/\rho^{0.5})\end{aligned}$$

Rectangular to Spherical

# What we are trying to achieve?

```
# modules: projector360.nn
from projector360.utils import projector
import torch.nn as nn

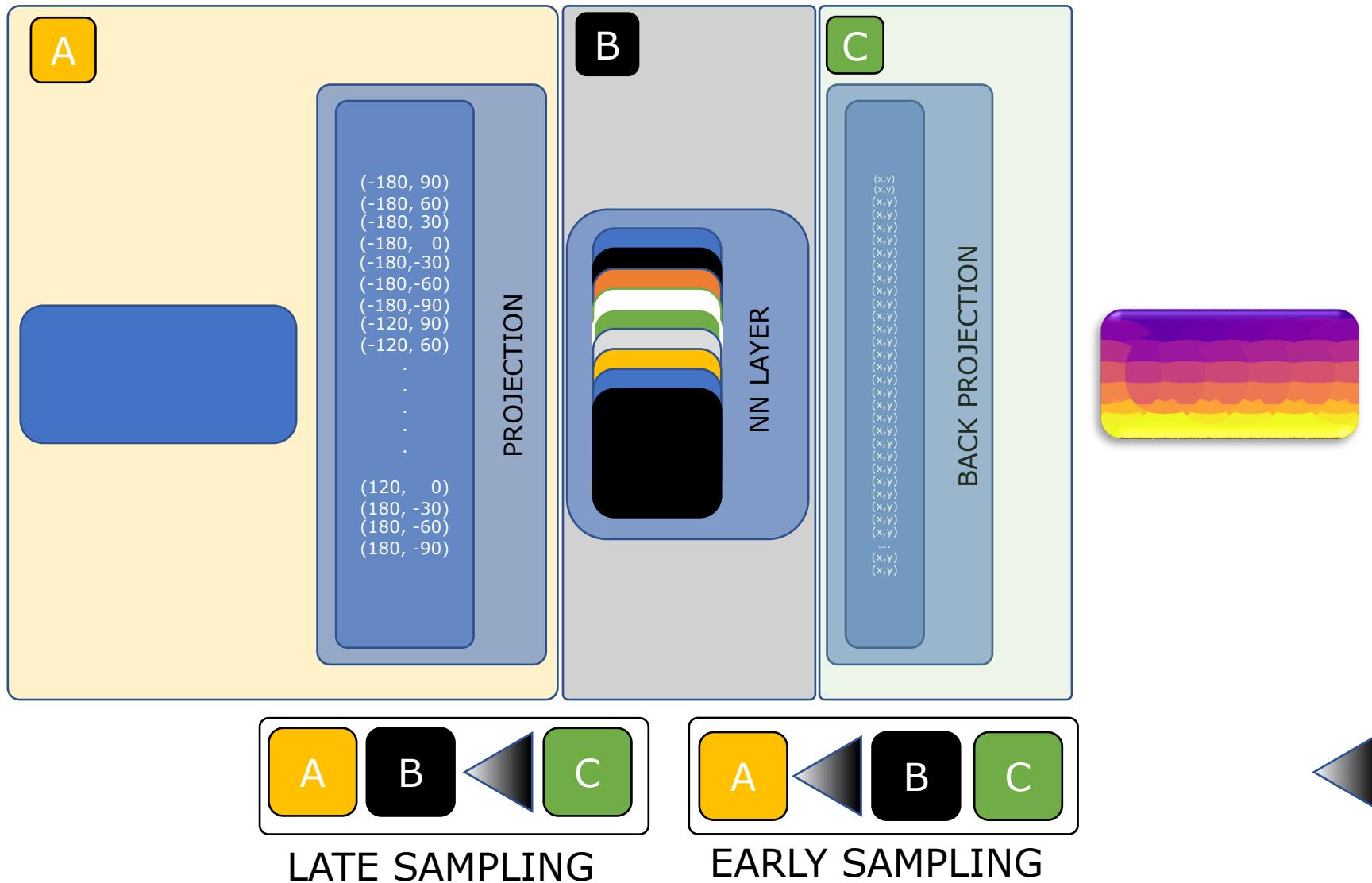
class Conv360(nn.Module):
    def __init__(self, *args, **kwargs):
        super(Conv360, self).__init__()
        self.proj_args = kwargs.pop('projector_args')
        self.conv360 = projector(
            nn.Conv2d(**kwargs),
            **self.proj_args
        )
    def initialize(self, weight, bias):
        self.conv360.weight.data = weight
        self.conv360.bias.data = bias

    def forward(self, x):
        return self.conv360(x)
...
```

```
# modules:perspectiveModel.py
from torch.nn import Conv2d
from torch.nn import MaxPool2d
from torch.nn.functional import interpolate
...
```

```
# modules:sphericalModel.py
from projector360.nn import Conv360 as Conv2d
from projector360.nn import MaxPool360 as MaxPool2d
from projector360.nn import Interpolate as interpolate
...
```

# Methodology



# Previous Implementation

- pypi: pip install py360convert
- <https://github.com/sunset1995/py360convert>

## Problems:

- Serial
- Not Differentiable

# Current Implementation

- pypi: pip install projector360 **[SOON]**
- <https://github.com/keshavsbandari/360projector>

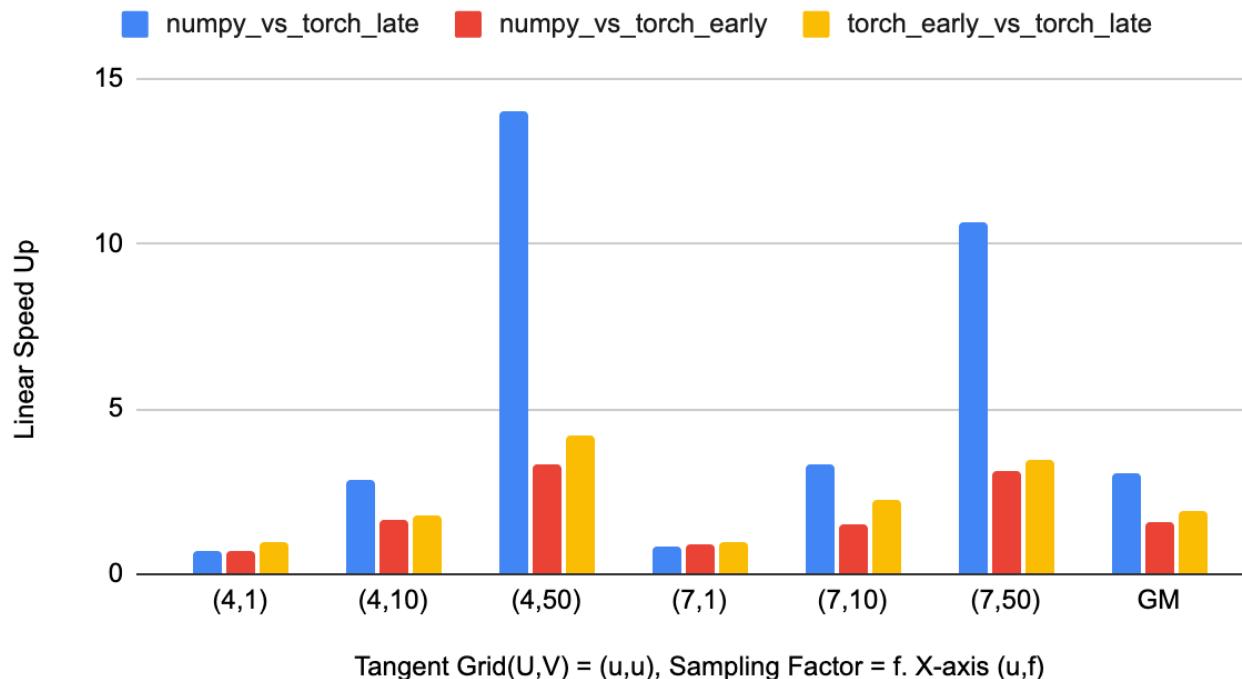
Benefit:

- Parallel
- Differentiable

# Performance Comparison

Speed Up Comparison on Absolute Runtime

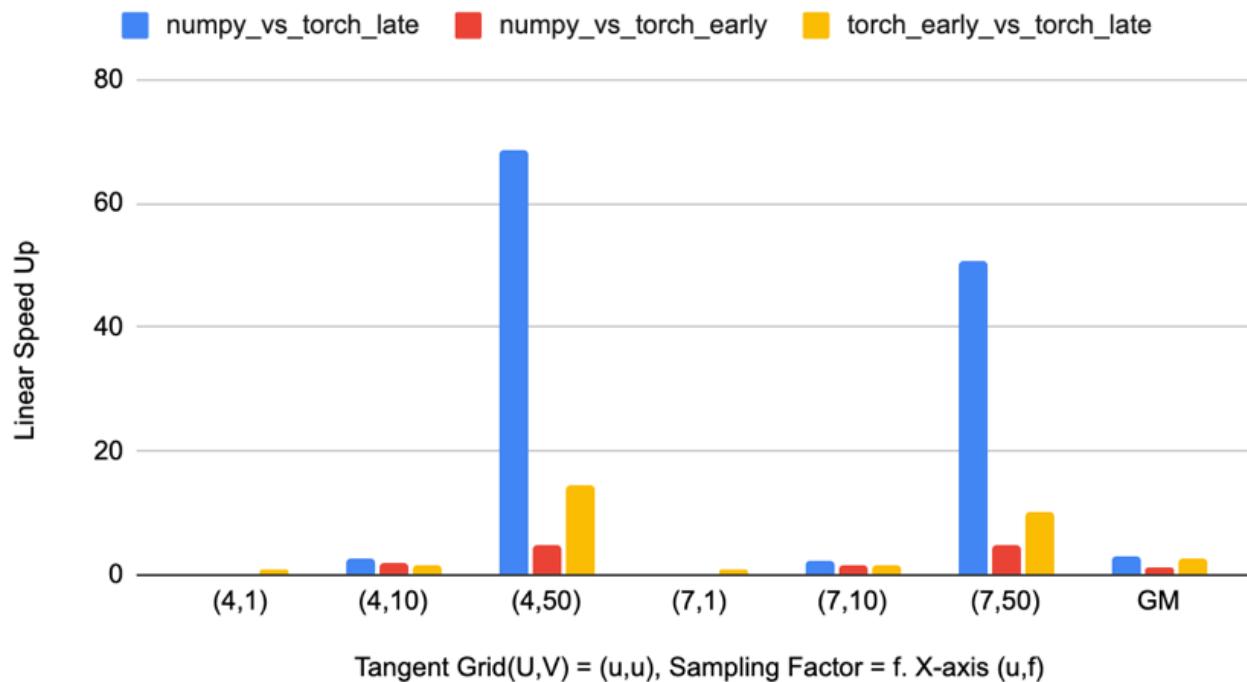
## Speed Up Comparison



# Performance Comparison

Speed Up Comparison on Initialization

Speed Up Comparison [Initialization]

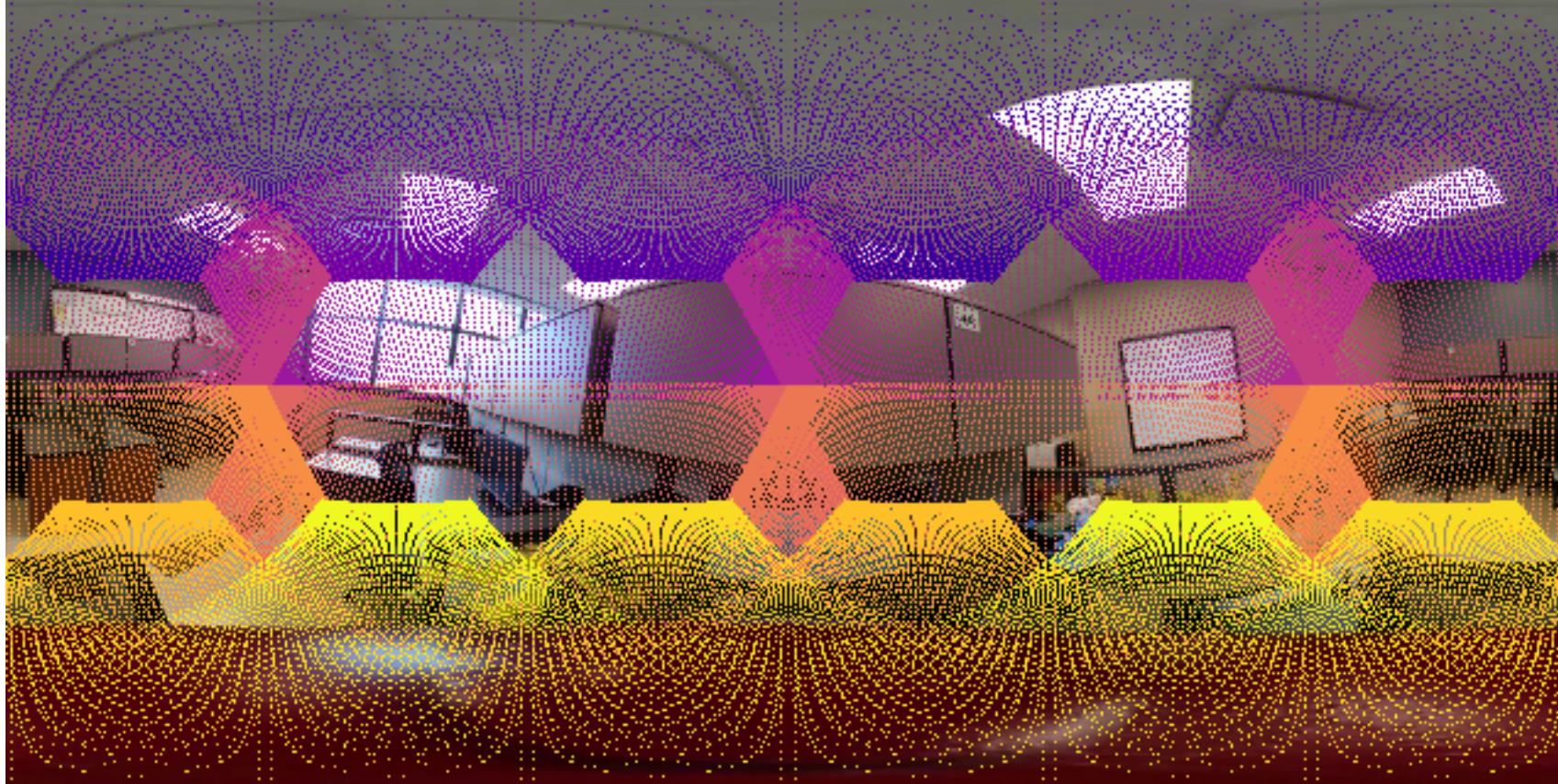


# Performance Comparison

## Execution Time Comparison

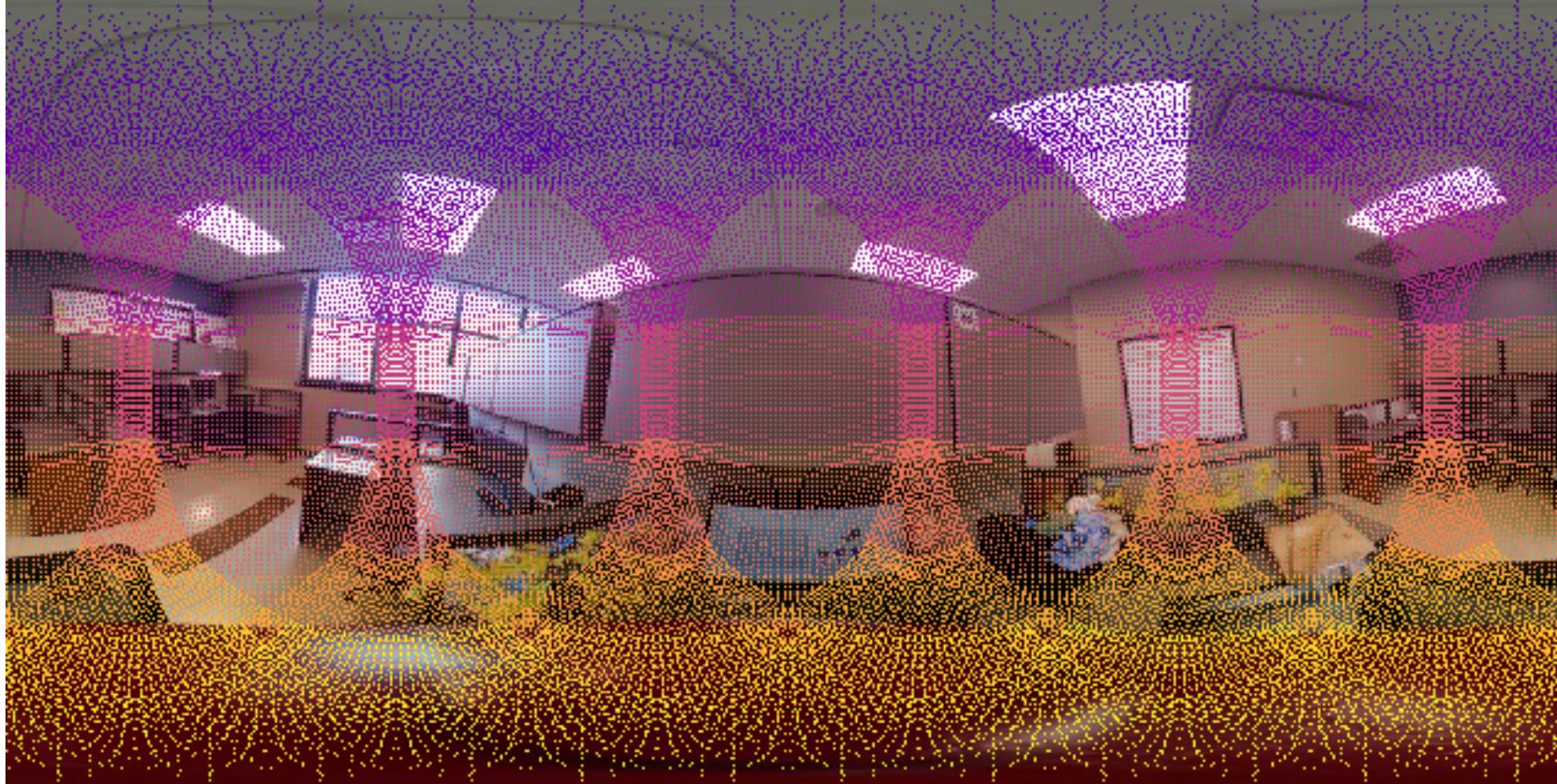
version	griduv	sample_factor	n	libraryload	totaltime	actualtime
numpy	4	1	25	0.2164945602	1.17	0.9535054398
numpy	4	10	25	3.819404602	13.758	9.938595398
numpy	4	50	25	93.51598191	322.069	228.5530181
numpy	7	1	25	0.3007824421	1.086	0.7852175579
numpy	7	10	25	3.36736393	11.729	8.36163607
numpy	7	50	25	81.4623847	278.657	197.1946153
gm_numpy				4.306634597		11.87504444
torch_early	4	1	25	1.29707551	2.602	1.30492449
torch_early	4	10	25	2.060480118	8.121	6.060519882
torch_early	4	50	25	20.00457716	88.57	68.56542284
torch_early	7	1	25	1.4298985	2.331	0.9011014996
torch_early	7	10	25	1.978047609	7.525	5.546952391
torch_early	7	50	25	16.48	80.323	63.843
gm_torch_early				3.6820841		7.464893962
torch_late	4	1	25	1.28	2.669	1.389
torch_late	4	10	25	1.436487913	4.888	3.451512087
torch_late	4	50	25	1.3652246	17.644	16.2787754
torch_late	7	1	25	1.672086954	2.62	0.9479130459
torch_late	7	10	25	1.377296448	3.872	2.494703552
torch_late	7	50	25	1.609186411	20.092	18.48281359
gm_torch_late				1.450222979		3.879848945

# Qualitative Results



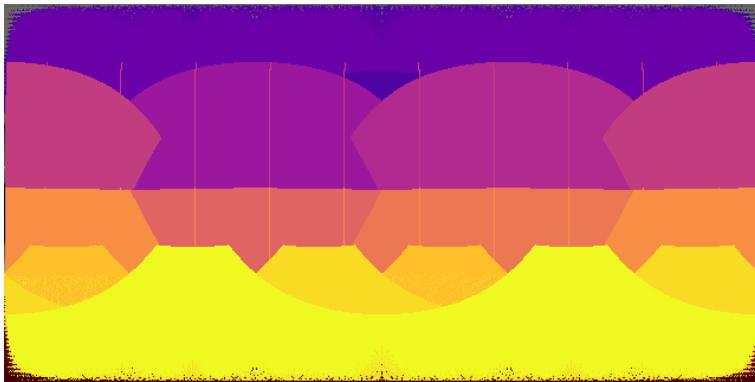
UV-GRID (4,4), sampling\_factor = 1, late

# Qualitative Results

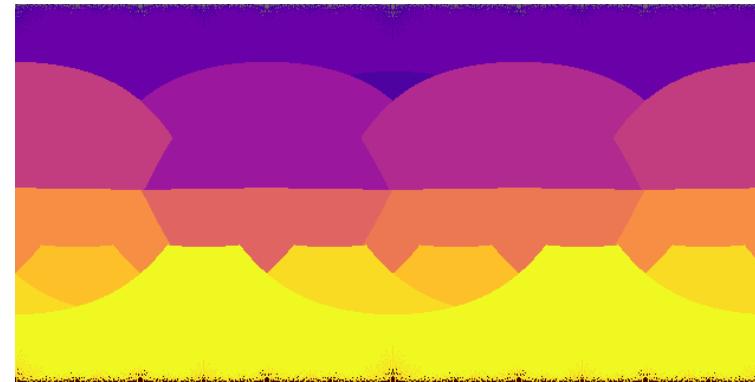


UV-GRID (7,7), sampling\_factor = 1, early

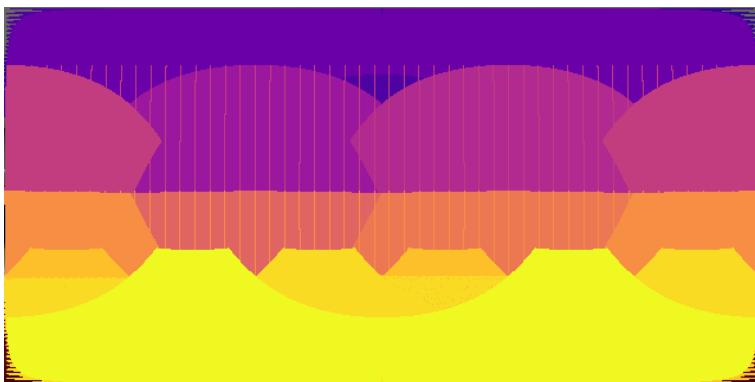
# Qualitative Results



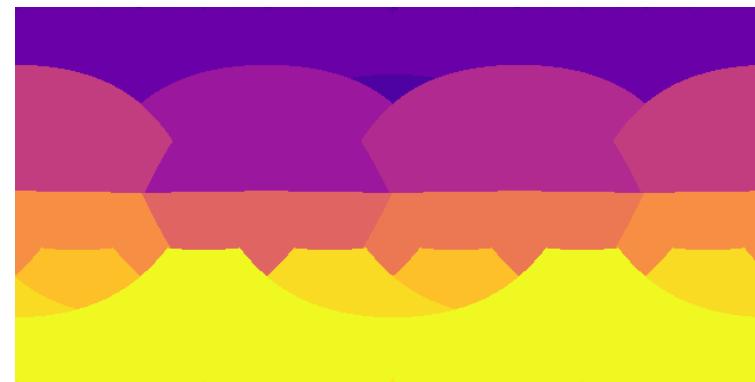
UV-GRID (4,4), sampling\_factor = 10, late



UV-GRID (4,4), sampling\_factor = 10, early

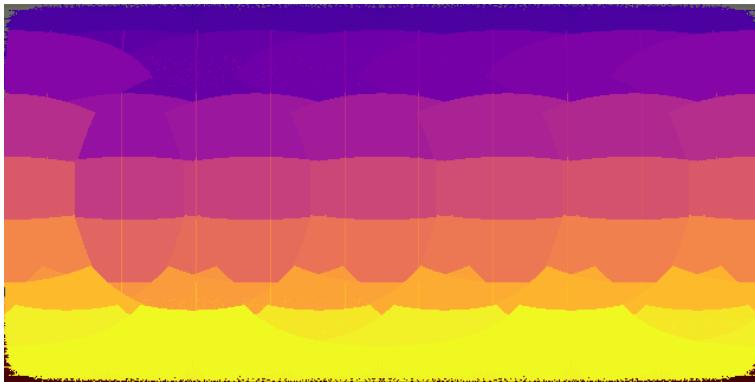


UV-GRID (4,4), sampling\_factor = 50, late

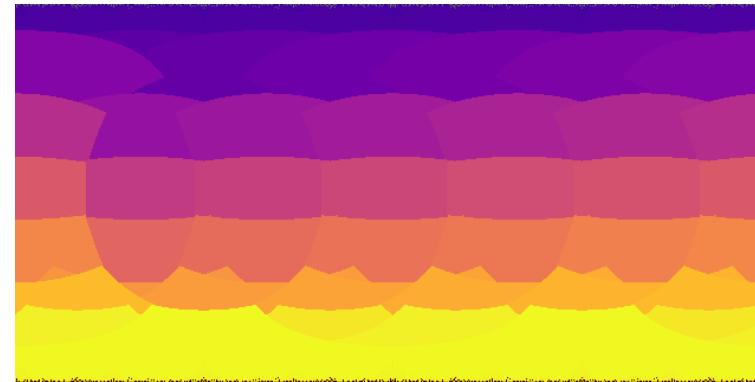


UV-GRID (4,4), sampling\_factor = 50, early

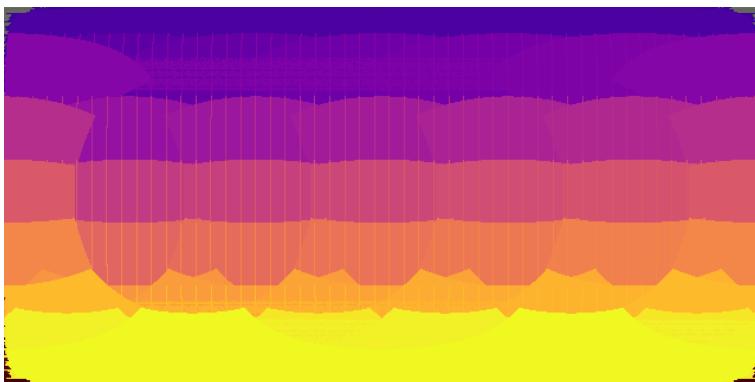
# Qualitative Results



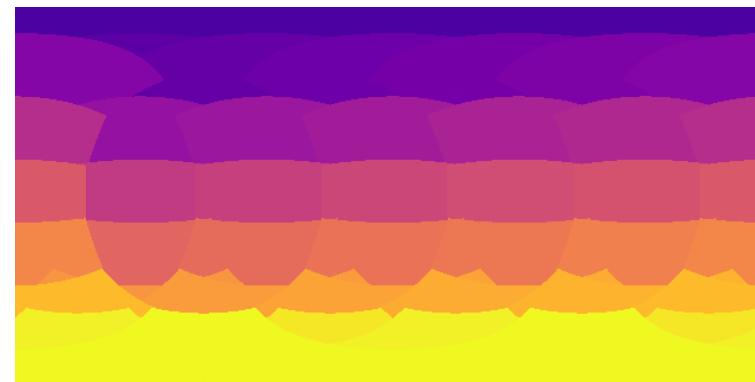
UV-GRID (7,7), sampling\_factor = 10, late



UV-GRID (7,7), sampling\_factor = 10, early



UV-GRID (7,7), sampling\_factor = 50, late

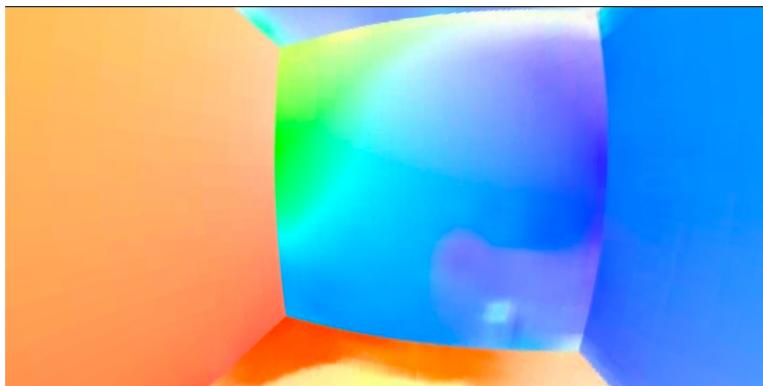


UV-GRID (7,7), sampling\_factor = 50, early

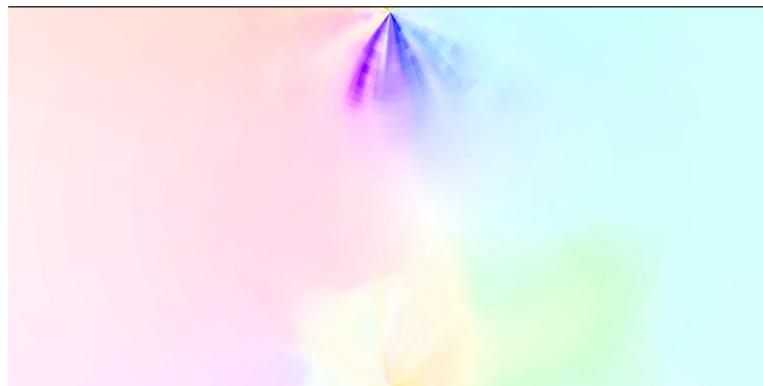
# Application Results

## \* What is optical flow?

*Displacement vectors associated with objects in consecutive frames of a video.*



Flow Using Cube-maps only



Optical flow using  
liteflownet360 + raft  
**STAGE-2**  
(Equirectangular Projection)



Optical flow using  
liteflownet360 + raft  
**FINAL**  
(Equirectangular Projection)

<https://arxiv.org/abs/2010.08045> : LiteFlowNet360  
<https://arxiv.org/abs/2003.12039> : Raft

# <Platform>

```
Architecture:          x86_64
CPU op-mode(s):       32-bit, 64-bit
Byte Order:           Little Endian
CPU(s):               12
On-line CPU(s) list: 0-11
Thread(s) per core:  2
Core(s) per socket:  6
Socket(s):            1
NUMA node(s):         1
Vendor ID:            GenuineIntel
CPU family:           6
Model:                158
Model name:           Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz
Stepping:              10
CPU MHz:              811.666
CPU max MHz:          4700.0000
CPU min MHz:          800.0000
BogoMIPS:              7399.70
Virtualization:       VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              12288K
NUMA node0 CPU(s):    0-11
+-----+
| NVIDIA-SMI 450.80.02     Driver Version: 450.80.02     CUDA Version: 11.0 |
|-----+-----+-----+
| GPU  Name      Persistence-M| Bus-Id      Disp.A  | Volatile Uncorr. ECC | | |
| Fan  Temp     Perf  Pwr:Usage/Cap|           Memory-Usage | GPU-Util  Compute M. |
|                               |               |              |           | MIG M. |
|-----+-----+-----+
|   0  GeForce GTX 108...  Off  | 00000000:01:00.0  On   |                  N/A | | |
| 20%   39C     P5    19W / 250W |           368MiB / 11175MiB |      1%     Default |
|                               |               |              |           | N/A |
+-----+-----+-----+
```

Pytorch 1.7  
Numpy 1.19.0  
Python 3.7

<Queries>

<Thank You!!>