

# Libdl

Adrià Font Calvarons

July 31, 2019



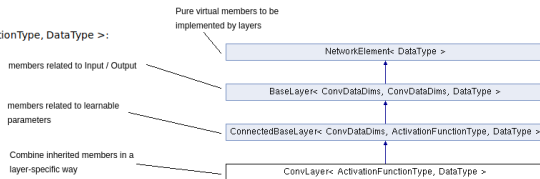
# Library Structure - Examples

## ConvLayer< ActivationFunctionType, DataType > Class Template Reference final

Conv Class for convolutional Layer elements. [More...](#)

```
#include <ConvLayer.h>
```

Inheritance diagram for ConvLayer< ActivationFunctionType, DataType >:



### Public Member Functions

**ConvLayer** (const size\_t aFilterHeight, const size\_t aFilterWidth, const size\_t aPaddingHeight, const size\_t aPaddingWidth, const size\_t aStride, const size\_t aInputSampleNumber, const **UpdateMethod** aUpdateMethod=UpdateMethod::NESTEROV)

**ConvLayer** (const size\_t aFilterHeight, const size\_t aFilterWidth, const size\_t aPaddingHeight, const size\_t aPaddingWidth, const size\_t aStride, const **ConvD** **UpdateMethod** aUpdateMethod=UpdateMethod::NESTEROV)

**ConvLayer** (const size\_t aFilterHeight, const size\_t aFilterWidth, const size\_t aPaddingHeight, const size\_t aPaddingWidth, const size\_t aStride, const **ConvD** const **UpdateMethod** aUpdateMethod=UpdateMethod::NESTEROV)

void **ForwardPass** () override

void **BackwardPass** () override

Public Member Functions inherited from **ConnectedBaseLayer**< **ConvDataDims**, **ActivationFunctionType**, **DataType** >

Public Member Functions inherited from **BaseLayer**< **ConvDataDims**, **ConvDataDims**, **DataType** >



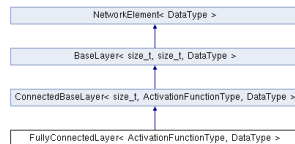
# Library Structure - Examples

## FullyConnectedLayer< ActivationFunctionType, DataType > Class Template Reference final

Fully connected layer for dense layer elements. [More...](#)

```
#include <FullyConnectedLayer.h>
```

Inheritance diagram for FullyConnectedLayer< ActivationFunctionType, DataType >:



## Public Member Functions

**FullyConnectedLayer** (const size\_t aInputDim, const size\_t aOutputDim, const **UpdateMethod** aUpdateMethod=UpdateMethod::NESTEROV)

void **ForwardPass** () override

void **BackwardPass** () override

► Public Member Functions inherited from **ConnectedBaseLayer**< size\_t, ActivationFunctionType, DataType >

► Public Member Functions inherited from **BaseLayer**< size\_t, size\_t, DataType >



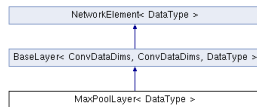
# Library Structure - Examples

## MaxPoolLayer< DataType > Class Template Reference final

MaxPool Layer. [More...](#)

```
#include <MaxPoolLayer.h>
```

Inheritance diagram for MaxPoolLayer< DataType >:



## Public Member Functions

**MaxPoolLayer** (const size\_t aInputDepth, const size\_t aInputHeight, const size\_t aInputWidth, const size\_t aPoolSize, const size\_t aStride, const size\_t aInputSampleNumber)

**MaxPoolLayer** (const `ConvDataDims` aInputDims, const size\_t aPoolSize, const size\_t aStride, const size\_t aInputSampleNumber)

void **ForwardPass** () override

void **BackwardPass** () override

► Public Member Functions inherited from `BaseLayer< ConvDataDims, ConvDataDims, DataType >`



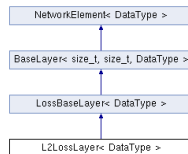
# Library Structure - Examples

## L2LossLayer< DataType > Class Template Reference final

L2 Loss Layer. [More...](#)

```
#include <L2LossLayer.h>
```

Inheritance diagram for L2LossLayer< DataType >:



## Public Member Functions

**L2LossLayer** (double aLossNormalizationFactor=1.0)

void **ForwardPass** () override

void **BackwardPass** () override

‣ Public Member Functions inherited from **LossBaseLayer< DataType >**

‣ Public Member Functions inherited from **BaseLayer< size\_t, size\_t, DataType >**



# Transposed Convolution (a.k.a Fractionally strided Convolution)

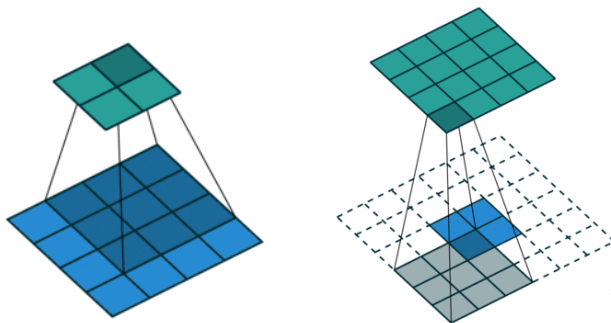
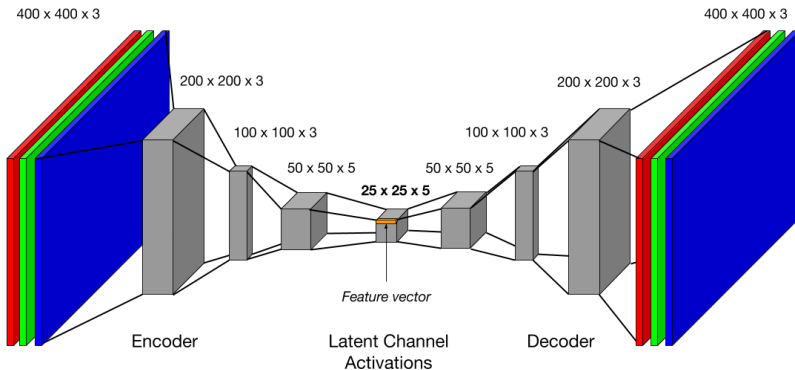


Figure: Conv vs TransposedConv ([github.com/vdumoulin/conv\\_arithmetic](https://github.com/vdumoulin/conv_arithmetic))



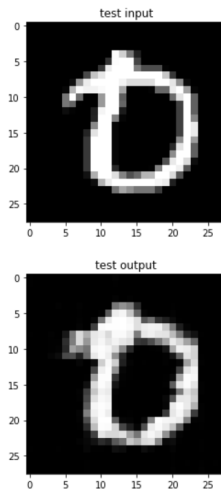
# Convolutional Autoencoder



**Figure:** Flaspohler, Genevieve, Nicholas Roy, and Yogesh Girdhar. "Feature discovery and visualization of robot mission data using convolutional autoencoders and Bayesian nonparametric topic models." IEEE, 2017.



# Convolutional Autoencoder



**Figure:** Results with Bottleneck feature vector of dimensions 7, 7, 4 (25%)





# Segmentation

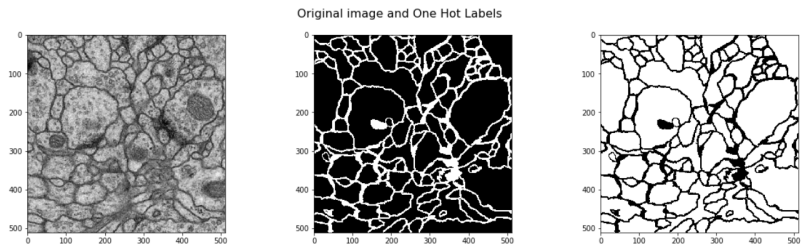


Figure: Sample of a 30 x 512 x 512 Stack (training/Testing)

# Segmentation

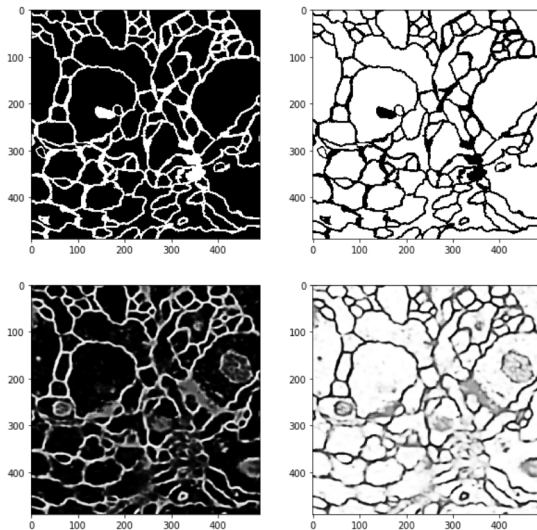


Figure: Top Ground truth, bottom results

# Segmentation

## Possible Improvements:

- ▶ Dedicated Loss function (eg: U-Net uses precomputed weight loss map).
- ▶ More Layers, more features (eg: U-Net has 1024 feature maps in "bottleneck").
- ▶ Skip Connections(Drozdzal, Michal, et al. "The importance of skip connections in biomedical image segmentation." Deep Learning and Data Labeling for Medical Applications. Springer, Cham, 2016. 179-187.)



# Thank you

Questions?

