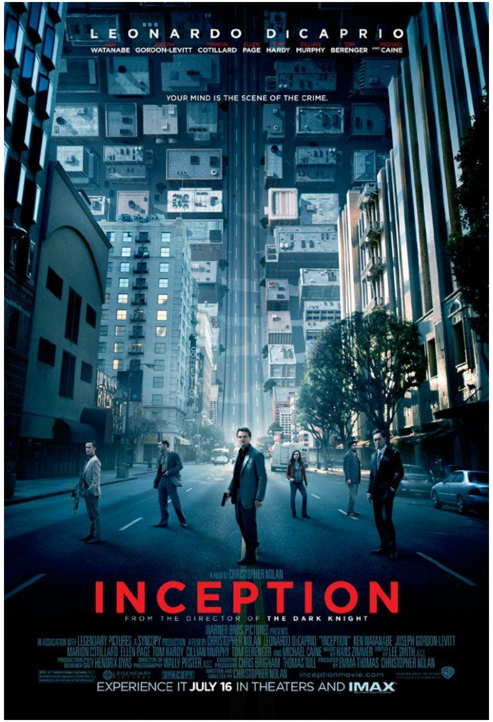
Inception Network Overview

David White
CS793

So, Leonardo DiCaprio dreams about dreaming...


LEONARDO DICAPRIO
KEN WATANABE JOSEPH GORDON-LEVITT MARION COTILLARD ELLEN PAGE TOM HARDY CILLIAN MURPHY TOM BERENGER AND MICHAEL CAINE
YOUR MIND IS THE SCENE OF THE CRIME.
A FILM BY CHRISTOPHER NOLAN
INCEPTION
FROM THE DIRECTOR OF THE DARK KNIGHT
WARNER BROS. PICTURES PRESENTS
IN ASSOCIATION WITH LEGENDARY PICTURES A SYNCOPY PRODUCTION A FILM BY CHRISTOPHER NOLAN LEONARDO DiCAPRIO "INCEPTION" KEN WATANABE JOSEPH GORDON-LEVITT
MARION COTILLARD ELLEN PAGE TOM HARDY CILLIAN MURPHY TOM BERENGER AND MICHAEL CAINE MUSIC BY HANS ZIMMER EDITED BY LEE SMITH, A.C.E.
PRODUCTION DESIGNER GUY HENDRIX DYAS DIRECTOR OF PHOTOGRAPHY WALLY PFISTER, A.S.C. EXECUTIVE PRODUCERS CHRIS BRIGHAM THOMAS TULL PRODUCED BY EMMA THOMAS CHRISTOPHER NOLAN
LEGENDARY PICTURES
SYNCOPY
WRITTEN AND DIRECTED BY CHRISTOPHER NOLAN
inceptionmovie.com
EXPERIENCE IT JULY 16 IN THEATERS AND IMAX

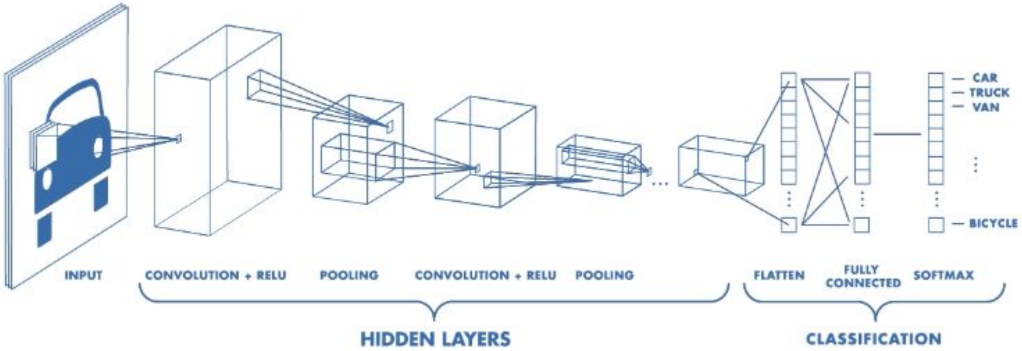
https://m.media-amazon.com/images/M/MV5BMjAxMzY3NjcxNF5BMl5BanBnXkFtZTcwNTI5OTM0Mw@@._V1_SY1000_CR0,0,675,1000_AL_.jpg
https://knowyourmeme.com/memes/we-need-to-go-deeper

Background

- ImageNet is a dataset consisting of millions hand-labeled images
- ILSVRC is an annual competition for the best image classifier on 1000 ImageNet classes,
    - Judged on top-5 error rate
- Deep (many-layered) Convolutional Neural Networks (CNNs) achieved surprisingly low error in 2012
    - AlexNet (15% over 25% in 2011)

Background

- Standard CNN structure up until 2014 was stacked convolutional layers, maybe max-pooling, then one or more fully-connected layers
- This has limits
  - Large memory footprint
  - Large computation demand
  - Prone to overfitting
  - Vanishing and exploding gradients
- AlexNet had ~60 million parameters



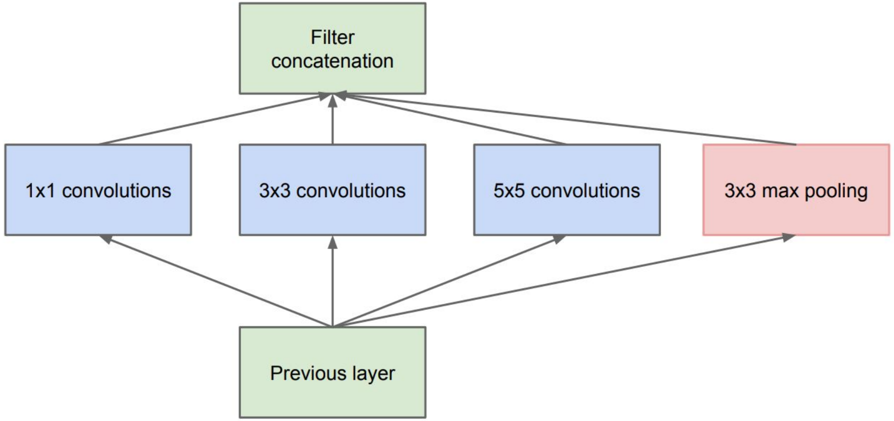
https://www.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html

Inception Motivation

- Dense connections are expensive
- Biological systems are sparse
- Sparsity can be exploited by clustering correlated outputs
    - Theoretical work by Arora et al.
- However
    - Non-uniform sparse matrix calculations are expensive
    - Dense calculations are very efficient
- Ultimately
    - “...[find] out how an optimal local sparse structure in a convolutional vision network can be approximated and covered by dense components”
    - “All we need is to find the optimal local construction and to repeat it spatially.”

Inception Module (naive)

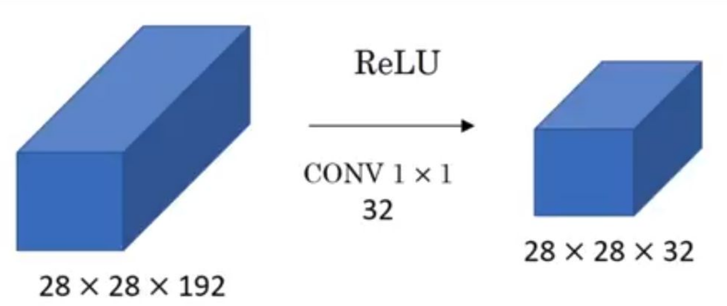
- Approximation of an optimal local sparse structure
- Process visual/spatial information at various scales and then aggregate
- This is a bit optimistic, computationally
    - 5x5 convolutions are especially expensive

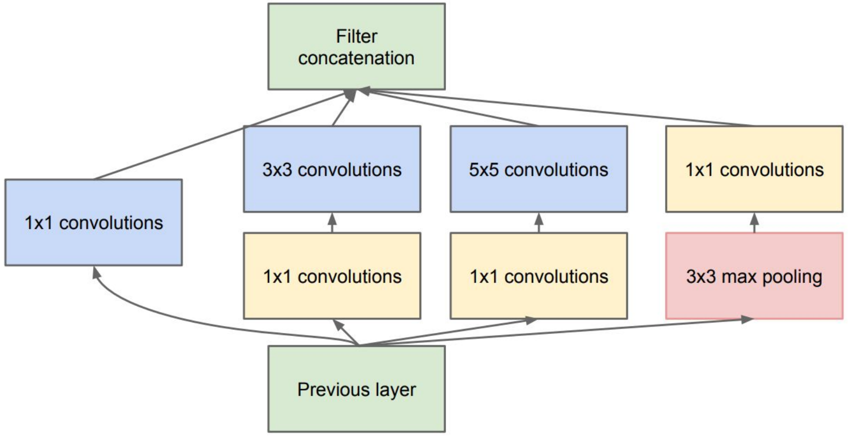


(a) Inception module, naïve version

Inception Module

- Dimension reduction is necessary and motivated (Network in network)
- Achieved with 1x1 convolutions
  - Think: **learned pooling in depth** instead of max/average pooling in height/width





(b) Inception module with dimension reductions

Originally: Lin et al. Network in network
Obtained from: Andrew Ng, Deeplearning.ai, https://youtu.be/c1RBQzKsDCk

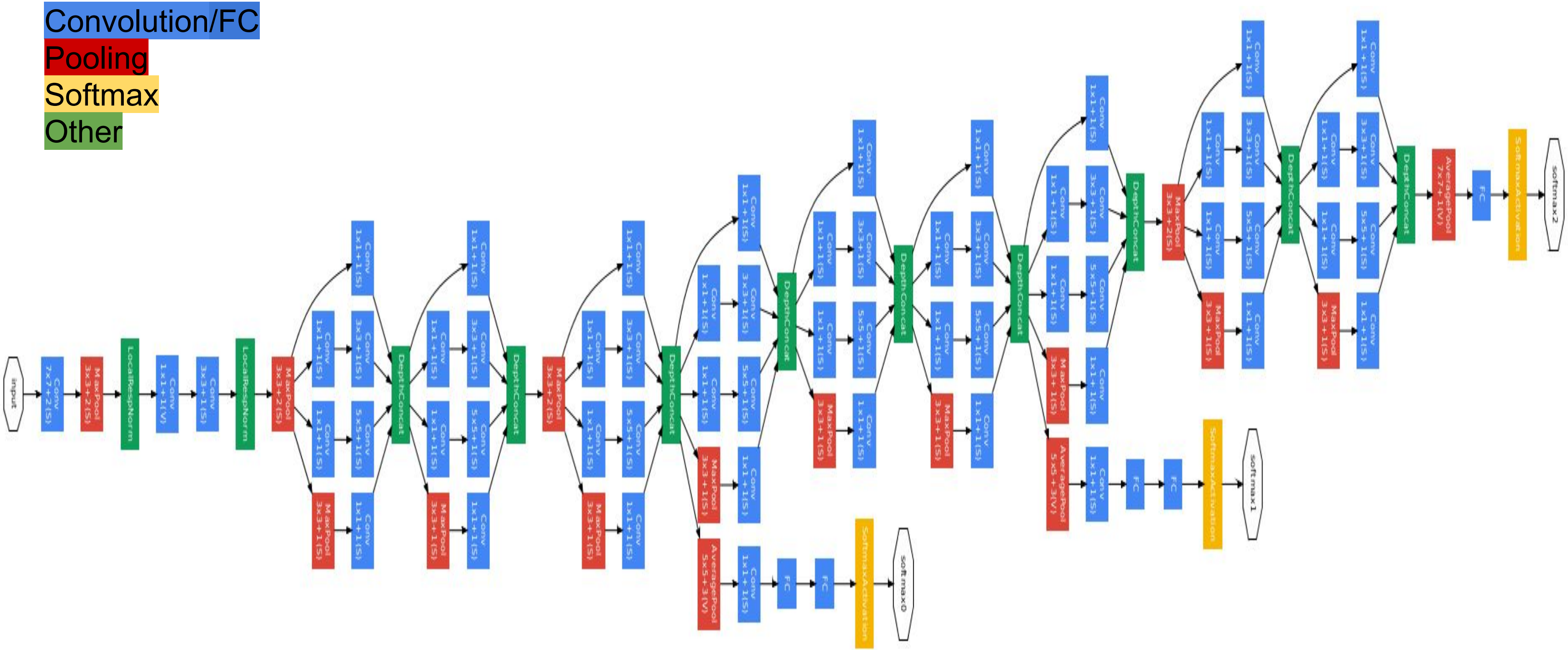
Full Inception-v1

Convolution/FC

Pooling

Softmax

Other



Full Inception-v1

- Stem: standard convolution, pooling, and normalization operations
- Body: 9 stacked inception modules
- Final Classifier: average pooling and single fc-layer
    - Shown to be slightly better than all fc-layers
- Auxiliary classifiers
    - encourage discrimination
    - provide regularization
    - discarded at inference
- 22 layers total, ~5 million parameters

Inception-v1 Results

- 7 Inception networks in an ensemble
  - differing only in sampling methodologies and random order
  - Averaged over many crops then over each network
- **6.67%** top-5 error rate, first place in ILSVRC2014

BN-Inception (Batch-Normalized)

- Addresses general “*internal covariate shift*”
    - Varied signal distribution across mini-batches
    - Broad benefits, including reducing time to train
- Applied to Inception-v1, besides the following
    - Removal of 5x5 convolutions in favor of two stacked 3x3
    - Increased learning rate and decay
    - Removal of dropout
    - Removal of L_2 weight regularization
    - Removal of Local Response Normalization
    - Reduced photometric distortions
    - 1 more 28x28 inception module
    - Changed pooling schemes

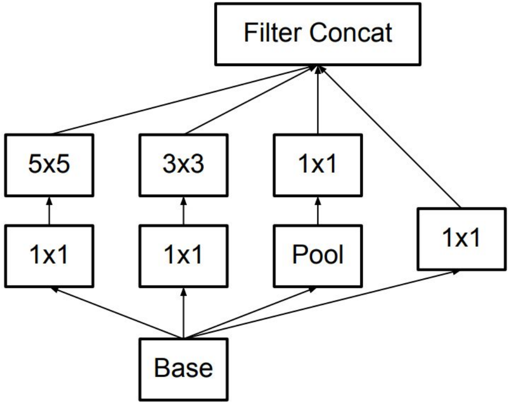


Figure 4. Original Inception module as described in [20].

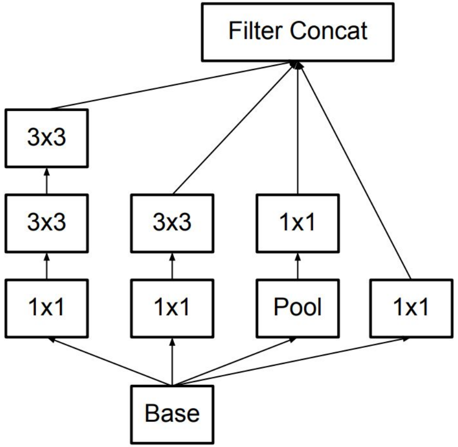


Figure 5. Inception modules where each 5×5 convolution is replaced by two 3×3 convolution, as suggested by principle 3 of Section 2.

BN-Inception Results

- Inception-v1 error reached in 14x fewer training steps
- **4.8% top-5** validation error in ensemble
  - Still 5x fewer steps
- Increased (?) computation cost
  - Vague

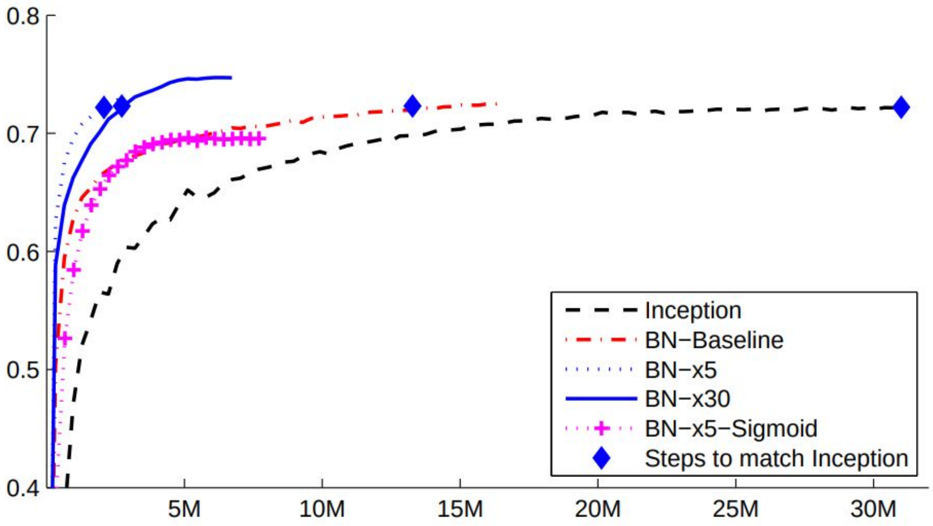


Figure 2: *Single crop validation accuracy of Inception and its batch-normalized variants, vs. the number of training steps.*

Inception-v2 and v3

- Iterative refinement
- Attempt to formalize goals and steps to achieve them
  - Avoid representational bottlenecks
  - Process high-dimensional representations locally
  - Perform spatial aggregation over low-dim embeddings
  - Balance the width and depth
  - **Ultimately: increase performance and decrease cost**
- Theory behind ideas backed by empirical evidence

Inception-v2 and v3

- Eliminate 5x5 in favor of two 3x3 convolutions (as per BN-Inception)
    - Decreases cost at no representational loss
- Assymetric convolutions over medium grid sizes (nx1 then 1xn)
    - Decreases cost at no representational loss (in medium case)
- Hybrid for high-dimension

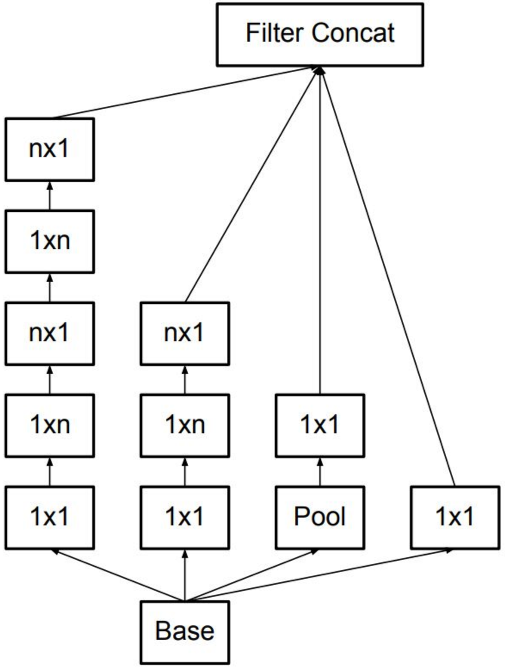


Figure 6. Inception modules after the factorization of the $n \times n$ convolutions. In our proposed architecture, we chose $n = 7$ for the 17×17 grid. (The filter sizes are picked using principle 3)

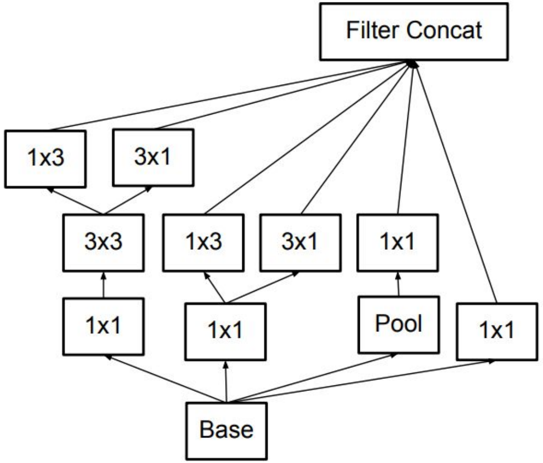


Figure 7. Inception modules with expanded the filter bank outputs. This architecture is used on the coarsest (8×8) grids to promote high dimensional representations, as suggested by principle 2 of Section 2. We are using this solution only on the coarsest grid, since that is the place where producing high dimensional sparse representation is the most critical as the ratio of local processing (by 1×1 convolutions) is increased compared to the spatial aggregation.

Inception-v2 and v3

- Reduction of auxiliary classifiers from 3 to 2
    - Not totally necessary in v1’s implementation
    - Effect is very small
- Further grid-size reduction
    - Parallel stride-2 pooling and stride-2 convolution

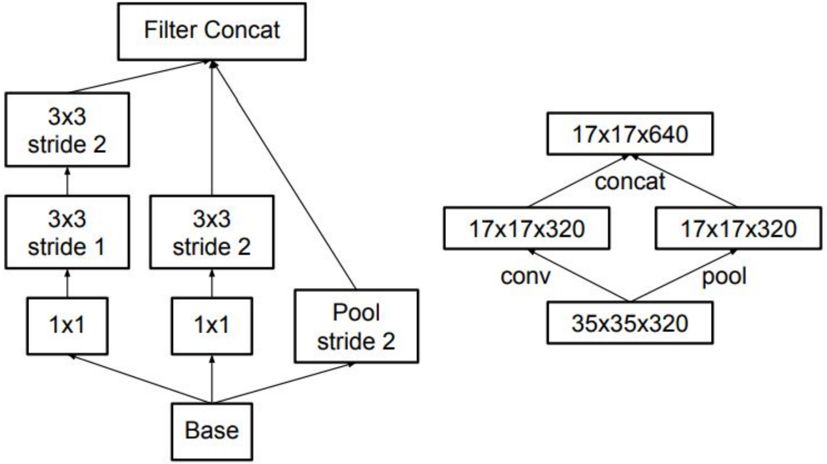
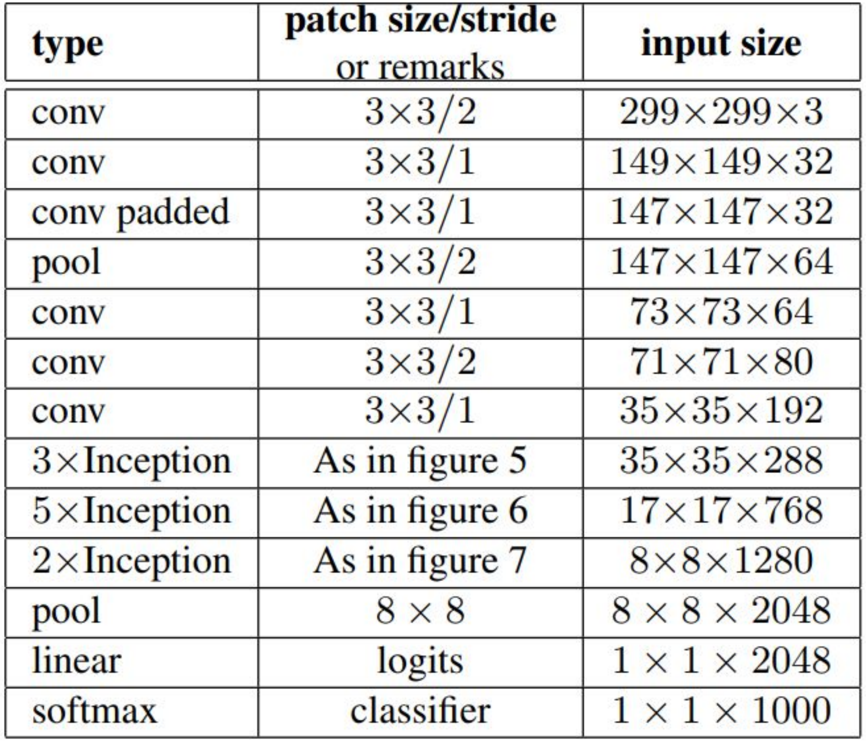


Figure 10. Inception module that reduces the grid-size while expands the filter banks. It is both cheap and avoids the representational bottleneck as is suggested by principle 1. The diagram on the right represents the same solution but from the perspective of grid sizes rather than the operations.

Inception-v2

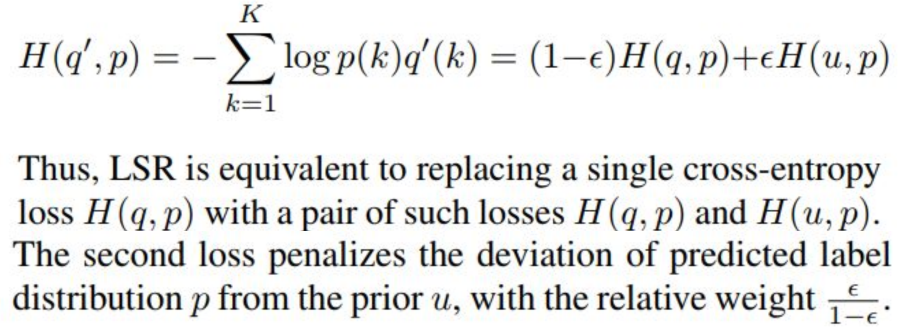
- Finally, these are all integrated into Inception-v2
- More layers means more accuracy
- Also more cost--but not prohibitively so



type	patch size/stride or remarks	input size
conv	3×3/2	299×299×3
conv	3×3/1	149×149×32
conv padded	3×3/1	147×147×32
pool	3×3/2	147×147×64
conv	3×3/1	73×73×64
conv	3×3/2	71×71×80
conv	3×3/1	35×35×192
3×Inception	As in figure 5	35×35×288
5×Inception	As in figure 6	17×17×768
2×Inception	As in figure 7	8×8×1280
pool	8 × 8	8 × 8 × 2048
linear	logits	1 × 1 × 2048
softmax	classifier	1 × 1 × 1000

Label Smoothing

- In brief: “a mechanism to regularize the classifier by estimating the effect of label-dropout during training”



$$H(q',p) = -\sum_{k=1}^{K} \log p(k) q'(k) = (1-\epsilon)H(q,p) + \epsilon H(u,p)$$

Thus, LSR is equivalent to replacing a single cross-entropy loss $H(q,p)$ with a pair of such losses $H(q,p)$ and $H(u,p)$. The second loss penalizes the deviation of predicted label distribution p from the prior u, with the relative weight $\frac{\epsilon}{1-\epsilon}$.

- Achieved a small (0.2%) improvement

Inception-v3

- Inception-v2 as previously described with
    - RMSProp
    - Label Smoothing
    - Auxiliary classifier’s fully-connected layer is batch-normalized
- Achieves **3.58% top-5** error on an ensemble of 4 models
    - Half the error of the original Inception-v1 ensemble (GoogLeNet).

Inception-v4

- “Inception-v3 had inherited a lot of the baggage of the earlier incarnations”
- See paper for schemas
- Little motivation for changes to pure Inception-v4 besides lost baggage

Inception-Resnet

- Residual connections in Inception-Resnet
  - Replaced filter concatenation
  - Required some dimensionality adjustments
  - Little effect on final accuracy (compared to similar-size pure Inception)
  - Decreased training time
  - Potential network “death”--training instability

Inception-v4 and Inception-Resnet Results

- Outperform previous iterations “by virtue of size alone”
- Residual connections consistently provide
  - Faster training
  - Slightly better prediction
- Ensemble of 3x Inception-Resnet(v2) and 1x Inception-v4 produces **3.08% top-5** error

Inception Network Overview

David White
CS793

Code!

http://cs.colostate.edu/~dwhite54/inception-v4-demo.tgz