#dontcare, Car, SUV, Truck-small, Truck-medium, Truck-large, Pedestrian, Bus, Van, GroupOfPeople, Bicyclist, Motorcycle, Trafficsignal-Green, Trafficsignal-Yellow, Trafficsignal-Red

# DetectNet network

# Data/Input layers

name: "DetectNet"

layer {

name: "train\_data"

type: "Data"

top: "data"

data\_param {

backend: LMDB

source: "examples/kitti/kitti\_train\_images.lmdb"

batch\_size: 10

}

include: { phase: TRAIN }

}

layer {

name: "train\_label"

type: "Data"

top: "label"

data\_param {

backend: LMDB

source: "examples/kitti/kitti\_train\_labels.lmdb"

batch\_size: 10

}

include: { phase: TRAIN }

}

layer {

name: "val\_data"

type: "Data"

top: "data"

data\_param {

backend: LMDB

source: "examples/kitti/kitti\_test\_images.lmdb"

batch\_size: 6

}

include: { phase: TEST stage: "val" }

}

layer {

name: "val\_label"

type: "Data"

top: "label"

data\_param {

backend: LMDB

source: "examples/kitti/kitti\_test\_labels.lmdb"

batch\_size: 6

}

include: { phase: TEST stage: "val" }

}

layer {

name: "deploy\_data"

type: "Input"

top: "data"

input\_param {

shape {

dim: 1

dim: 3

dim: 1088

dim: 1920

}

}

include: { phase: TEST not\_stage: "val" }

}

# Data transformation layers

layer {

name: "train\_transform"

type: "DetectNetTransformation"

bottom: "data"

bottom: "label"

top: "transformed\_data"

top: "transformed\_label"

detectnet\_groundtruth\_param: {

stride: 16

scale\_cvg: 0.4

gridbox\_type: GRIDBOX\_MIN

coverage\_type: RECTANGULAR

min\_cvg\_len: 20

obj\_norm: true

image\_size\_x: 1920

image\_size\_y: 1088

crop\_bboxes: true

object\_class: { src: 1 dst: 0} # cars -> 0

object\_class: { src: 2 dst: 1} # SUV -> 1

object\_class: { src: 3 dst: 2} # Truck-small -> 2

object\_class: { src: 4 dst: 3} # Truck-medium -> 3

object\_class: { src: 5 dst: 4} # Truck-large -> 4

object\_class: { src: 6 dst: 5} # Pedestrian -> 5

object\_class: { src: 7 dst: 6} # Bus -> 6

object\_class: { src: 8 dst: 7} # Van -> 7

object\_class: { src: 9 dst: 8} # GroupOfPeople -> 8

object\_class: { src: 10 dst: 9} # Bicyclist -> 9

object\_class: { src: 11 dst: 10} # Motorcycle -> 10

object\_class: { src: 12 dst: 11} # Trafficsignal-Green -> 11

object\_class: { src: 13 dst: 12} # Trafficsignal-Yellow -> 12

object\_class: { src: 14 dst: 13} # Trafficsignal-Red -> 13

}

detectnet\_augmentation\_param: {

crop\_prob: 1

shift\_x: 32

shift\_y: 32

flip\_prob: 0.5

rotation\_prob: 0

max\_rotate\_degree: 5

scale\_prob: 0.4

scale\_min: 0.8

scale\_max: 1.2

hue\_rotation\_prob: 0.8

hue\_rotation: 30

desaturation\_prob: 0.8

desaturation\_max: 0.8

}

transform\_param: {

mean\_value: 127

}

include: { phase: TRAIN }

}

layer {

name: "val\_transform"

type: "DetectNetTransformation"

bottom: "data"

bottom: "label"

top: "transformed\_data"

top: "transformed\_label"

detectnet\_groundtruth\_param: {

stride: 16

scale\_cvg: 0.4

gridbox\_type: GRIDBOX\_MIN

coverage\_type: RECTANGULAR

min\_cvg\_len: 20

obj\_norm: true

image\_size\_x: 1920

image\_size\_y: 1088

crop\_bboxes: false

object\_class: { src: 1 dst: 0} # cars -> 0

object\_class: { src: 2 dst: 1} # SUV -> 1

object\_class: { src: 3 dst: 2} # Truck-small -> 2

object\_class: { src: 4 dst: 3} # Truck-medium -> 3

object\_class: { src: 5 dst: 4} # Truck-large -> 4

object\_class: { src: 6 dst: 5} # Pedestrian -> 5

object\_class: { src: 7 dst: 6} # Bus -> 6

object\_class: { src: 8 dst: 7} # Van -> 7

object\_class: { src: 9 dst: 8} # GroupOfPeople -> 8

object\_class: { src: 10 dst: 9} # Bicyclist -> 9

object\_class: { src: 11 dst: 10} # Motorcycle -> 10

object\_class: { src: 12 dst: 11} # Trafficsignal-Green -> 11

object\_class: { src: 13 dst: 12} # Trafficsignal-Yellow -> 12

object\_class: { src: 14 dst: 13} # Trafficsignal-Red -> 13

}

transform\_param: {

mean\_value: 127

}

include: { phase: TEST stage: "val" }

}

layer {

name: "deploy\_transform"

type: "Power"

bottom: "data"

top: "transformed\_data"

power\_param {

shift: -127

}

include: { phase: TEST not\_stage: "val" }

}

# Label conversion layers

layer {

name: "slice-label"

type: "Slice"

bottom: "transformed\_label"

top: "foreground-label"

top: "bbox-label"

top: "size-label"

top: "obj-label"

top: "coverage-label"

slice\_param {

slice\_dim: 1

slice\_point: 1

slice\_point: 5

slice\_point: 7

slice\_point: 8

}

include { phase: TRAIN }

include { phase: TEST stage: "val" }

}

layer {

name: "coverage-block"

type: "Concat"

bottom: "foreground-label"

bottom: "foreground-label"

bottom: "foreground-label"

bottom: "foreground-label"

top: "coverage-block"

concat\_param {

concat\_dim: 1

}

include { phase: TRAIN }

include { phase: TEST stage: "val" }

}

layer {

name: "size-block"

type: "Concat"

bottom: "size-label"

bottom: "size-label"

top: "size-block"

concat\_param {

concat\_dim: 1

}

include { phase: TRAIN }

include { phase: TEST stage: "val" }

}

layer {

name: "obj-block"

type: "Concat"

bottom: "obj-label"

bottom: "obj-label"

bottom: "obj-label"

bottom: "obj-label"

top: "obj-block"

concat\_param {

concat\_dim: 1

}

include { phase: TRAIN }

include { phase: TEST stage: "val" }

}

layer {

name: "bb-label-norm"

type: "Eltwise"

bottom: "bbox-label"

bottom: "size-block"

top: "bbox-label-norm"

eltwise\_param {

operation: PROD

}

include { phase: TRAIN }

include { phase: TEST stage: "val" }

}

layer {

name: "bb-obj-norm"

type: "Eltwise"

bottom: "bbox-label-norm"

bottom: "obj-block"

top: "bbox-obj-label-norm"

eltwise\_param {

operation: PROD

}

include { phase: TRAIN }

include { phase: TEST stage: "val" }

}

######################################################################

# Start of convolutional network

######################################################################

layer {

name: "conv1/7x7\_s2"

type: "Convolution"

bottom: "transformed\_data"

top: "conv1/7x7\_s2"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 64

pad: 3

kernel\_size: 7

stride: 2

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "conv1/relu\_7x7"

type: "ReLU"

bottom: "conv1/7x7\_s2"

top: "conv1/7x7\_s2"

}

layer {

name: "pool1/3x3\_s2"

type: "Pooling"

bottom: "conv1/7x7\_s2"

top: "pool1/3x3\_s2"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 2

}

}

layer {

name: "pool1/norm1"

type: "LRN"

bottom: "pool1/3x3\_s2"

top: "pool1/norm1"

lrn\_param {

local\_size: 5

alpha: 0.0001

beta: 0.75

}

}

layer {

name: "conv2/3x3\_reduce"

type: "Convolution"

bottom: "pool1/norm1"

top: "conv2/3x3\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 64

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "conv2/relu\_3x3\_reduce"

type: "ReLU"

bottom: "conv2/3x3\_reduce"

top: "conv2/3x3\_reduce"

}

layer {

name: "conv2/3x3"

type: "Convolution"

bottom: "conv2/3x3\_reduce"

top: "conv2/3x3"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 192

pad: 1

kernel\_size: 3

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "conv2/relu\_3x3"

type: "ReLU"

bottom: "conv2/3x3"

top: "conv2/3x3"

}

layer {

name: "conv2/norm2"

type: "LRN"

bottom: "conv2/3x3"

top: "conv2/norm2"

lrn\_param {

local\_size: 5

alpha: 0.0001

beta: 0.75

}

}

layer {

name: "pool2/3x3\_s2"

type: "Pooling"

bottom: "conv2/norm2"

top: "pool2/3x3\_s2"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 2

}

}

layer {

name: "inception\_3a/1x1"

type: "Convolution"

bottom: "pool2/3x3\_s2"

top: "inception\_3a/1x1"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 64

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3a/relu\_1x1"

type: "ReLU"

bottom: "inception\_3a/1x1"

top: "inception\_3a/1x1"

}

layer {

name: "inception\_3a/3x3\_reduce"

type: "Convolution"

bottom: "pool2/3x3\_s2"

top: "inception\_3a/3x3\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 96

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.09

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3a/relu\_3x3\_reduce"

type: "ReLU"

bottom: "inception\_3a/3x3\_reduce"

top: "inception\_3a/3x3\_reduce"

}

layer {

name: "inception\_3a/3x3"

type: "Convolution"

bottom: "inception\_3a/3x3\_reduce"

top: "inception\_3a/3x3"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 128

pad: 1

kernel\_size: 3

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3a/relu\_3x3"

type: "ReLU"

bottom: "inception\_3a/3x3"

top: "inception\_3a/3x3"

}

layer {

name: "inception\_3a/5x5\_reduce"

type: "Convolution"

bottom: "pool2/3x3\_s2"

top: "inception\_3a/5x5\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 16

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.2

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3a/relu\_5x5\_reduce"

type: "ReLU"

bottom: "inception\_3a/5x5\_reduce"

top: "inception\_3a/5x5\_reduce"

}

layer {

name: "inception\_3a/5x5"

type: "Convolution"

bottom: "inception\_3a/5x5\_reduce"

top: "inception\_3a/5x5"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 32

pad: 2

kernel\_size: 5

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3a/relu\_5x5"

type: "ReLU"

bottom: "inception\_3a/5x5"

top: "inception\_3a/5x5"

}

layer {

name: "inception\_3a/pool"

type: "Pooling"

bottom: "pool2/3x3\_s2"

top: "inception\_3a/pool"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 1

pad: 1

}

}

layer {

name: "inception\_3a/pool\_proj"

type: "Convolution"

bottom: "inception\_3a/pool"

top: "inception\_3a/pool\_proj"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 32

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3a/relu\_pool\_proj"

type: "ReLU"

bottom: "inception\_3a/pool\_proj"

top: "inception\_3a/pool\_proj"

}

layer {

name: "inception\_3a/output"

type: "Concat"

bottom: "inception\_3a/1x1"

bottom: "inception\_3a/3x3"

bottom: "inception\_3a/5x5"

bottom: "inception\_3a/pool\_proj"

top: "inception\_3a/output"

}

layer {

name: "inception\_3b/1x1"

type: "Convolution"

bottom: "inception\_3a/output"

top: "inception\_3b/1x1"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 128

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3b/relu\_1x1"

type: "ReLU"

bottom: "inception\_3b/1x1"

top: "inception\_3b/1x1"

}

layer {

name: "inception\_3b/3x3\_reduce"

type: "Convolution"

bottom: "inception\_3a/output"

top: "inception\_3b/3x3\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 128

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.09

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3b/relu\_3x3\_reduce"

type: "ReLU"

bottom: "inception\_3b/3x3\_reduce"

top: "inception\_3b/3x3\_reduce"

}

layer {

name: "inception\_3b/3x3"

type: "Convolution"

bottom: "inception\_3b/3x3\_reduce"

top: "inception\_3b/3x3"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 192

pad: 1

kernel\_size: 3

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3b/relu\_3x3"

type: "ReLU"

bottom: "inception\_3b/3x3"

top: "inception\_3b/3x3"

}

layer {

name: "inception\_3b/5x5\_reduce"

type: "Convolution"

bottom: "inception\_3a/output"

top: "inception\_3b/5x5\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 32

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.2

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3b/relu\_5x5\_reduce"

type: "ReLU"

bottom: "inception\_3b/5x5\_reduce"

top: "inception\_3b/5x5\_reduce"

}

layer {

name: "inception\_3b/5x5"

type: "Convolution"

bottom: "inception\_3b/5x5\_reduce"

top: "inception\_3b/5x5"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 96

pad: 2

kernel\_size: 5

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3b/relu\_5x5"

type: "ReLU"

bottom: "inception\_3b/5x5"

top: "inception\_3b/5x5"

}

layer {

name: "inception\_3b/pool"

type: "Pooling"

bottom: "inception\_3a/output"

top: "inception\_3b/pool"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 1

pad: 1

}

}

layer {

name: "inception\_3b/pool\_proj"

type: "Convolution"

bottom: "inception\_3b/pool"

top: "inception\_3b/pool\_proj"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 64

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_3b/relu\_pool\_proj"

type: "ReLU"

bottom: "inception\_3b/pool\_proj"

top: "inception\_3b/pool\_proj"

}

layer {

name: "inception\_3b/output"

type: "Concat"

bottom: "inception\_3b/1x1"

bottom: "inception\_3b/3x3"

bottom: "inception\_3b/5x5"

bottom: "inception\_3b/pool\_proj"

top: "inception\_3b/output"

}

layer {

name: "pool3/3x3\_s2"

type: "Pooling"

bottom: "inception\_3b/output"

top: "pool3/3x3\_s2"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 2

}

}

layer {

name: "inception\_4a/1x1"

type: "Convolution"

bottom: "pool3/3x3\_s2"

top: "inception\_4a/1x1"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 192

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4a/relu\_1x1"

type: "ReLU"

bottom: "inception\_4a/1x1"

top: "inception\_4a/1x1"

}

layer {

name: "inception\_4a/3x3\_reduce"

type: "Convolution"

bottom: "pool3/3x3\_s2"

top: "inception\_4a/3x3\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 96

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.09

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4a/relu\_3x3\_reduce"

type: "ReLU"

bottom: "inception\_4a/3x3\_reduce"

top: "inception\_4a/3x3\_reduce"

}

layer {

name: "inception\_4a/3x3"

type: "Convolution"

bottom: "inception\_4a/3x3\_reduce"

top: "inception\_4a/3x3"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 208

pad: 1

kernel\_size: 3

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4a/relu\_3x3"

type: "ReLU"

bottom: "inception\_4a/3x3"

top: "inception\_4a/3x3"

}

layer {

name: "inception\_4a/5x5\_reduce"

type: "Convolution"

bottom: "pool3/3x3\_s2"

top: "inception\_4a/5x5\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 16

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.2

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4a/relu\_5x5\_reduce"

type: "ReLU"

bottom: "inception\_4a/5x5\_reduce"

top: "inception\_4a/5x5\_reduce"

}

layer {

name: "inception\_4a/5x5"

type: "Convolution"

bottom: "inception\_4a/5x5\_reduce"

top: "inception\_4a/5x5"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 48

pad: 2

kernel\_size: 5

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4a/relu\_5x5"

type: "ReLU"

bottom: "inception\_4a/5x5"

top: "inception\_4a/5x5"

}

layer {

name: "inception\_4a/pool"

type: "Pooling"

bottom: "pool3/3x3\_s2"

top: "inception\_4a/pool"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 1

pad: 1

}

}

layer {

name: "inception\_4a/pool\_proj"

type: "Convolution"

bottom: "inception\_4a/pool"

top: "inception\_4a/pool\_proj"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 64

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4a/relu\_pool\_proj"

type: "ReLU"

bottom: "inception\_4a/pool\_proj"

top: "inception\_4a/pool\_proj"

}

layer {

name: "inception\_4a/output"

type: "Concat"

bottom: "inception\_4a/1x1"

bottom: "inception\_4a/3x3"

bottom: "inception\_4a/5x5"

bottom: "inception\_4a/pool\_proj"

top: "inception\_4a/output"

}

layer {

name: "inception\_4b/1x1"

type: "Convolution"

bottom: "inception\_4a/output"

top: "inception\_4b/1x1"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 160

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4b/relu\_1x1"

type: "ReLU"

bottom: "inception\_4b/1x1"

top: "inception\_4b/1x1"

}

layer {

name: "inception\_4b/3x3\_reduce"

type: "Convolution"

bottom: "inception\_4a/output"

top: "inception\_4b/3x3\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 112

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.09

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4b/relu\_3x3\_reduce"

type: "ReLU"

bottom: "inception\_4b/3x3\_reduce"

top: "inception\_4b/3x3\_reduce"

}

layer {

name: "inception\_4b/3x3"

type: "Convolution"

bottom: "inception\_4b/3x3\_reduce"

top: "inception\_4b/3x3"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 224

pad: 1

kernel\_size: 3

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4b/relu\_3x3"

type: "ReLU"

bottom: "inception\_4b/3x3"

top: "inception\_4b/3x3"

}

layer {

name: "inception\_4b/5x5\_reduce"

type: "Convolution"

bottom: "inception\_4a/output"

top: "inception\_4b/5x5\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 24

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.2

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4b/relu\_5x5\_reduce"

type: "ReLU"

bottom: "inception\_4b/5x5\_reduce"

top: "inception\_4b/5x5\_reduce"

}

layer {

name: "inception\_4b/5x5"

type: "Convolution"

bottom: "inception\_4b/5x5\_reduce"

top: "inception\_4b/5x5"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 64

pad: 2

kernel\_size: 5

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4b/relu\_5x5"

type: "ReLU"

bottom: "inception\_4b/5x5"

top: "inception\_4b/5x5"

}

layer {

name: "inception\_4b/pool"

type: "Pooling"

bottom: "inception\_4a/output"

top: "inception\_4b/pool"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 1

pad: 1

}

}

layer {

name: "inception\_4b/pool\_proj"

type: "Convolution"

bottom: "inception\_4b/pool"

top: "inception\_4b/pool\_proj"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 64

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4b/relu\_pool\_proj"

type: "ReLU"

bottom: "inception\_4b/pool\_proj"

top: "inception\_4b/pool\_proj"

}

layer {

name: "inception\_4b/output"

type: "Concat"

bottom: "inception\_4b/1x1"

bottom: "inception\_4b/3x3"

bottom: "inception\_4b/5x5"

bottom: "inception\_4b/pool\_proj"

top: "inception\_4b/output"

}

layer {

name: "inception\_4c/1x1"

type: "Convolution"

bottom: "inception\_4b/output"

top: "inception\_4c/1x1"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 128

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4c/relu\_1x1"

type: "ReLU"

bottom: "inception\_4c/1x1"

top: "inception\_4c/1x1"

}

layer {

name: "inception\_4c/3x3\_reduce"

type: "Convolution"

bottom: "inception\_4b/output"

top: "inception\_4c/3x3\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 128

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.09

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4c/relu\_3x3\_reduce"

type: "ReLU"

bottom: "inception\_4c/3x3\_reduce"

top: "inception\_4c/3x3\_reduce"

}

layer {

name: "inception\_4c/3x3"

type: "Convolution"

bottom: "inception\_4c/3x3\_reduce"

top: "inception\_4c/3x3"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 256

pad: 1

kernel\_size: 3

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4c/relu\_3x3"

type: "ReLU"

bottom: "inception\_4c/3x3"

top: "inception\_4c/3x3"

}

layer {

name: "inception\_4c/5x5\_reduce"

type: "Convolution"

bottom: "inception\_4b/output"

top: "inception\_4c/5x5\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 24

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.2

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4c/relu\_5x5\_reduce"

type: "ReLU"

bottom: "inception\_4c/5x5\_reduce"

top: "inception\_4c/5x5\_reduce"

}

layer {

name: "inception\_4c/5x5"

type: "Convolution"

bottom: "inception\_4c/5x5\_reduce"

top: "inception\_4c/5x5"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 64

pad: 2

kernel\_size: 5

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4c/relu\_5x5"

type: "ReLU"

bottom: "inception\_4c/5x5"

top: "inception\_4c/5x5"

}

layer {

name: "inception\_4c/pool"

type: "Pooling"

bottom: "inception\_4b/output"

top: "inception\_4c/pool"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 1

pad: 1

}

}

layer {

name: "inception\_4c/pool\_proj"

type: "Convolution"

bottom: "inception\_4c/pool"

top: "inception\_4c/pool\_proj"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 64

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4c/relu\_pool\_proj"

type: "ReLU"

bottom: "inception\_4c/pool\_proj"

top: "inception\_4c/pool\_proj"

}

layer {

name: "inception\_4c/output"

type: "Concat"

bottom: "inception\_4c/1x1"

bottom: "inception\_4c/3x3"

bottom: "inception\_4c/5x5"

bottom: "inception\_4c/pool\_proj"

top: "inception\_4c/output"

}

layer {

name: "inception\_4d/1x1"

type: "Convolution"

bottom: "inception\_4c/output"

top: "inception\_4d/1x1"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 112

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4d/relu\_1x1"

type: "ReLU"

bottom: "inception\_4d/1x1"

top: "inception\_4d/1x1"

}

layer {

name: "inception\_4d/3x3\_reduce"

type: "Convolution"

bottom: "inception\_4c/output"

top: "inception\_4d/3x3\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 144

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4d/relu\_3x3\_reduce"

type: "ReLU"

bottom: "inception\_4d/3x3\_reduce"

top: "inception\_4d/3x3\_reduce"

}

layer {

name: "inception\_4d/3x3"

type: "Convolution"

bottom: "inception\_4d/3x3\_reduce"

top: "inception\_4d/3x3"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 288

pad: 1

kernel\_size: 3

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4d/relu\_3x3"

type: "ReLU"

bottom: "inception\_4d/3x3"

top: "inception\_4d/3x3"

}

layer {

name: "inception\_4d/5x5\_reduce"

type: "Convolution"

bottom: "inception\_4c/output"

top: "inception\_4d/5x5\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 32

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4d/relu\_5x5\_reduce"

type: "ReLU"

bottom: "inception\_4d/5x5\_reduce"

top: "inception\_4d/5x5\_reduce"

}

layer {

name: "inception\_4d/5x5"

type: "Convolution"

bottom: "inception\_4d/5x5\_reduce"

top: "inception\_4d/5x5"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 64

pad: 2

kernel\_size: 5

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4d/relu\_5x5"

type: "ReLU"

bottom: "inception\_4d/5x5"

top: "inception\_4d/5x5"

}

layer {

name: "inception\_4d/pool"

type: "Pooling"

bottom: "inception\_4c/output"

top: "inception\_4d/pool"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 1

pad: 1

}

}

layer {

name: "inception\_4d/pool\_proj"

type: "Convolution"

bottom: "inception\_4d/pool"

top: "inception\_4d/pool\_proj"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 64

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4d/relu\_pool\_proj"

type: "ReLU"

bottom: "inception\_4d/pool\_proj"

top: "inception\_4d/pool\_proj"

}

layer {

name: "inception\_4d/output"

type: "Concat"

bottom: "inception\_4d/1x1"

bottom: "inception\_4d/3x3"

bottom: "inception\_4d/5x5"

bottom: "inception\_4d/pool\_proj"

top: "inception\_4d/output"

}

layer {

name: "inception\_4e/1x1"

type: "Convolution"

bottom: "inception\_4d/output"

top: "inception\_4e/1x1"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 256

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4e/relu\_1x1"

type: "ReLU"

bottom: "inception\_4e/1x1"

top: "inception\_4e/1x1"

}

layer {

name: "inception\_4e/3x3\_reduce"

type: "Convolution"

bottom: "inception\_4d/output"

top: "inception\_4e/3x3\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 160

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.09

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4e/relu\_3x3\_reduce"

type: "ReLU"

bottom: "inception\_4e/3x3\_reduce"

top: "inception\_4e/3x3\_reduce"

}

layer {

name: "inception\_4e/3x3"

type: "Convolution"

bottom: "inception\_4e/3x3\_reduce"

top: "inception\_4e/3x3"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 320

pad: 1

kernel\_size: 3

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4e/relu\_3x3"

type: "ReLU"

bottom: "inception\_4e/3x3"

top: "inception\_4e/3x3"

}

layer {

name: "inception\_4e/5x5\_reduce"

type: "Convolution"

bottom: "inception\_4d/output"

top: "inception\_4e/5x5\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 32

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.2

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4e/relu\_5x5\_reduce"

type: "ReLU"

bottom: "inception\_4e/5x5\_reduce"

top: "inception\_4e/5x5\_reduce"

}

layer {

name: "inception\_4e/5x5"

type: "Convolution"

bottom: "inception\_4e/5x5\_reduce"

top: "inception\_4e/5x5"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 128

pad: 2

kernel\_size: 5

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4e/relu\_5x5"

type: "ReLU"

bottom: "inception\_4e/5x5"

top: "inception\_4e/5x5"

}

layer {

name: "inception\_4e/pool"

type: "Pooling"

bottom: "inception\_4d/output"

top: "inception\_4e/pool"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 1

pad: 1

}

}

layer {

name: "inception\_4e/pool\_proj"

type: "Convolution"

bottom: "inception\_4e/pool"

top: "inception\_4e/pool\_proj"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 128

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_4e/relu\_pool\_proj"

type: "ReLU"

bottom: "inception\_4e/pool\_proj"

top: "inception\_4e/pool\_proj"

}

layer {

name: "inception\_4e/output"

type: "Concat"

bottom: "inception\_4e/1x1"

bottom: "inception\_4e/3x3"

bottom: "inception\_4e/5x5"

bottom: "inception\_4e/pool\_proj"

top: "inception\_4e/output"

}

layer {

name: "inception\_5a/1x1"

type: "Convolution"

bottom: "inception\_4e/output"

top: "inception\_5a/1x1"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 256

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5a/relu\_1x1"

type: "ReLU"

bottom: "inception\_5a/1x1"

top: "inception\_5a/1x1"

}

layer {

name: "inception\_5a/3x3\_reduce"

type: "Convolution"

bottom: "inception\_4e/output"

top: "inception\_5a/3x3\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 160

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.09

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5a/relu\_3x3\_reduce"

type: "ReLU"

bottom: "inception\_5a/3x3\_reduce"

top: "inception\_5a/3x3\_reduce"

}

layer {

name: "inception\_5a/3x3"

type: "Convolution"

bottom: "inception\_5a/3x3\_reduce"

top: "inception\_5a/3x3"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 320

pad: 1

kernel\_size: 3

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5a/relu\_3x3"

type: "ReLU"

bottom: "inception\_5a/3x3"

top: "inception\_5a/3x3"

}

layer {

name: "inception\_5a/5x5\_reduce"

type: "Convolution"

bottom: "inception\_4e/output"

top: "inception\_5a/5x5\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 32

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.2

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5a/relu\_5x5\_reduce"

type: "ReLU"

bottom: "inception\_5a/5x5\_reduce"

top: "inception\_5a/5x5\_reduce"

}

layer {

name: "inception\_5a/5x5"

type: "Convolution"

bottom: "inception\_5a/5x5\_reduce"

top: "inception\_5a/5x5"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 128

pad: 2

kernel\_size: 5

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5a/relu\_5x5"

type: "ReLU"

bottom: "inception\_5a/5x5"

top: "inception\_5a/5x5"

}

layer {

name: "inception\_5a/pool"

type: "Pooling"

bottom: "inception\_4e/output"

top: "inception\_5a/pool"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 1

pad: 1

}

}

layer {

name: "inception\_5a/pool\_proj"

type: "Convolution"

bottom: "inception\_5a/pool"

top: "inception\_5a/pool\_proj"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 128

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5a/relu\_pool\_proj"

type: "ReLU"

bottom: "inception\_5a/pool\_proj"

top: "inception\_5a/pool\_proj"

}

layer {

name: "inception\_5a/output"

type: "Concat"

bottom: "inception\_5a/1x1"

bottom: "inception\_5a/3x3"

bottom: "inception\_5a/5x5"

bottom: "inception\_5a/pool\_proj"

top: "inception\_5a/output"

}

layer {

name: "inception\_5b/1x1"

type: "Convolution"

bottom: "inception\_5a/output"

top: "inception\_5b/1x1"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 384

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5b/relu\_1x1"

type: "ReLU"

bottom: "inception\_5b/1x1"

top: "inception\_5b/1x1"

}

layer {

name: "inception\_5b/3x3\_reduce"

type: "Convolution"

bottom: "inception\_5a/output"

top: "inception\_5b/3x3\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 1

decay\_mult: 0

}

convolution\_param {

num\_output: 192

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5b/relu\_3x3\_reduce"

type: "ReLU"

bottom: "inception\_5b/3x3\_reduce"

top: "inception\_5b/3x3\_reduce"

}

layer {

name: "inception\_5b/3x3"

type: "Convolution"

bottom: "inception\_5b/3x3\_reduce"

top: "inception\_5b/3x3"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 384

pad: 1

kernel\_size: 3

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5b/relu\_3x3"

type: "ReLU"

bottom: "inception\_5b/3x3"

top: "inception\_5b/3x3"

}

layer {

name: "inception\_5b/5x5\_reduce"

type: "Convolution"

bottom: "inception\_5a/output"

top: "inception\_5b/5x5\_reduce"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 48

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5b/relu\_5x5\_reduce"

type: "ReLU"

bottom: "inception\_5b/5x5\_reduce"

top: "inception\_5b/5x5\_reduce"

}

layer {

name: "inception\_5b/5x5"

type: "Convolution"

bottom: "inception\_5b/5x5\_reduce"

top: "inception\_5b/5x5"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 128

pad: 2

kernel\_size: 5

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5b/relu\_5x5"

type: "ReLU"

bottom: "inception\_5b/5x5"

top: "inception\_5b/5x5"

}

layer {

name: "inception\_5b/pool"

type: "Pooling"

bottom: "inception\_5a/output"

top: "inception\_5b/pool"

pooling\_param {

pool: MAX

kernel\_size: 3

stride: 1

pad: 1

}

}

layer {

name: "inception\_5b/pool\_proj"

type: "Convolution"

bottom: "inception\_5b/pool"

top: "inception\_5b/pool\_proj"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 128

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.1

}

bias\_filler {

type: "constant"

value: 0.2

}

}

}

layer {

name: "inception\_5b/relu\_pool\_proj"

type: "ReLU"

bottom: "inception\_5b/pool\_proj"

top: "inception\_5b/pool\_proj"

}

layer {

name: "inception\_5b/output"

type: "Concat"

bottom: "inception\_5b/1x1"

bottom: "inception\_5b/3x3"

bottom: "inception\_5b/5x5"

bottom: "inception\_5b/pool\_proj"

top: "inception\_5b/output"

}

layer {

name: "pool5/drop\_s1"

type: "Dropout"

bottom: "inception\_5b/output"

top: "pool5/drop\_s1"

dropout\_param {

dropout\_ratio: 0.4

}

}

layer {

name: "cvg/classifier"

type: "Convolution"

bottom: "pool5/drop\_s1"

top: "cvg/classifier"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 14

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.

}

}

}

layer {

name: "coverage/sig"

type: "Sigmoid"

bottom: "cvg/classifier"

top: "coverage"

}

layer {

name: "bbox/regressor"

type: "Convolution"

bottom: "pool5/drop\_s1"

top: "bboxes"

param {

lr\_mult: 1

decay\_mult: 1

}

param {

lr\_mult: 2

decay\_mult: 0

}

convolution\_param {

num\_output: 4

kernel\_size: 1

weight\_filler {

type: "xavier"

std: 0.03

}

bias\_filler {

type: "constant"

value: 0.

}

}

}

######################################################################

# End of convolutional network

######################################################################

# Convert bboxes

layer {

name: "bbox\_mask"

type: "Eltwise"

bottom: "bboxes"

bottom: "coverage-block"

top: "bboxes-masked"

eltwise\_param {

operation: PROD

}

include { phase: TRAIN }

include { phase: TEST stage: "val" }

}

layer {

name: "bbox-norm"

type: "Eltwise"

bottom: "bboxes-masked"

bottom: "size-block"

top: "bboxes-masked-norm"

eltwise\_param {

operation: PROD

}

include { phase: TRAIN }

include { phase: TEST stage: "val" }

}

layer {

name: "bbox-obj-norm"

type: "Eltwise"

bottom: "bboxes-masked-norm"

bottom: "obj-block"

top: "bboxes-obj-masked-norm"

eltwise\_param {

operation: PROD

}

include { phase: TRAIN }

include { phase: TEST stage: "val" }

}

# Loss layers

layer {

name: "bbox\_loss"

type: "L1Loss"

bottom: "bboxes-obj-masked-norm"

bottom: "bbox-obj-label-norm"

top: "loss\_bbox"

loss\_weight: 2

include { phase: TRAIN }

include { phase: TEST stage: "val" }

}

layer {

name: "coverage\_loss"

type: "EuclideanLoss"

bottom: "coverage"

bottom: "coverage-label"

top: "loss\_coverage"

include { phase: TRAIN }

include { phase: TEST stage: "val" }

}

# Cluster bboxes

layer {

type: 'Python'

name: 'cluster'

bottom: 'coverage'

bottom: 'bboxes'

top: 'bbox-list-class0'

top: 'bbox-list-class1'

top: 'bbox-list-class2'

top: 'bbox-list-class3'

top: 'bbox-list-class4'

top: 'bbox-list-class5'

top: 'bbox-list-class6'

top: 'bbox-list-class7'

top: 'bbox-list-class8'

top: 'bbox-list-class9'

top: 'bbox-list-class10'

top: 'bbox-list-class11'

top: 'bbox-list-class12'

top: 'bbox-list-class13'

python\_param {

module: 'caffe.layers.detectnet.clustering'

layer: 'ClusterDetections'

param\_str : '1920, 1088, 16, 0.6, 3, 0.02, 22, 14'

}

include: { phase: TEST }

}

# Calculate mean average precision

layer {

type: 'Python'

name: 'cluster\_gt'

bottom: 'coverage-label'

bottom: 'bbox-label'

top: 'bbox-list-label-class0'

top: 'bbox-list-label-class1'

top: 'bbox-list-label-class2'

top: 'bbox-list-label-class3'

top: 'bbox-list-label-class4'

top: 'bbox-list-label-class5'

top: 'bbox-list-label-class6'

top: 'bbox-list-label-class7'

top: 'bbox-list-label-class8'

top: 'bbox-list-label-class9'

top: 'bbox-list-label-class10'

top: 'bbox-list-label-class11'

top: 'bbox-list-label-class12'

top: 'bbox-list-label-class13'

python\_param {

module: 'caffe.layers.detectnet.clustering'

layer: 'ClusterGroundtruth'

param\_str : '1920, 1088, 16, 14'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class0'

bottom: 'bbox-list-label-class0'

bottom: 'bbox-list-class0'

top: 'bbox-list-scored-class0'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class0'

bottom: 'bbox-list-scored-class0'

top: 'mAP-class0'

top: 'precision-class0'

top: 'recall-class0'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class1'

bottom: 'bbox-list-label-class1'

bottom: 'bbox-list-class1'

top: 'bbox-list-scored-class1'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class1'

bottom: 'bbox-list-scored-class1'

top: 'mAP-class1'

top: 'precision-class1'

top: 'recall-class1'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class2'

bottom: 'bbox-list-label-class2'

bottom: 'bbox-list-class2'

top: 'bbox-list-scored-class2'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class2'

bottom: 'bbox-list-scored-class2'

top: 'mAP-class2'

top: 'precision-class2'

top: 'recall-class2'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class3'

bottom: 'bbox-list-label-class3'

bottom: 'bbox-list-class3'

top: 'bbox-list-scored-class3'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class3'

bottom: 'bbox-list-scored-class3'

top: 'mAP-class3'

top: 'precision-class3'

top: 'recall-class3'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class4'

bottom: 'bbox-list-label-class4'

bottom: 'bbox-list-class4'

top: 'bbox-list-scored-class4'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class4'

bottom: 'bbox-list-scored-class4'

top: 'mAP-class4'

top: 'precision-class4'

top: 'recall-class4'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class5'

bottom: 'bbox-list-label-class5'

bottom: 'bbox-list-class5'

top: 'bbox-list-scored-class5'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class5'

bottom: 'bbox-list-scored-class5'

top: 'mAP-class5'

top: 'precision-class5'

top: 'recall-class5'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class6'

bottom: 'bbox-list-label-class6'

bottom: 'bbox-list-class6'

top: 'bbox-list-scored-class6'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class6'

bottom: 'bbox-list-scored-class6'

top: 'mAP-class6'

top: 'precision-class6'

top: 'recall-class6'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class7'

bottom: 'bbox-list-label-class7'

bottom: 'bbox-list-class7'

top: 'bbox-list-scored-class7'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class7'

bottom: 'bbox-list-scored-class7'

top: 'mAP-class7'

top: 'precision-class7'

top: 'recall-class7'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class8'

bottom: 'bbox-list-label-class8'

bottom: 'bbox-list-class8'

top: 'bbox-list-scored-class8'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class8'

bottom: 'bbox-list-scored-class8'

top: 'mAP-class8'

top: 'precision-class8'

top: 'recall-class8'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class9'

bottom: 'bbox-list-label-class9'

bottom: 'bbox-list-class9'

top: 'bbox-list-scored-class9'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class9'

bottom: 'bbox-list-scored-class9'

top: 'mAP-class9'

top: 'precision-class9'

top: 'recall-class9'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class10'

bottom: 'bbox-list-label-class10'

bottom: 'bbox-list-class10'

top: 'bbox-list-scored-class10'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class10'

bottom: 'bbox-list-scored-class10'

top: 'mAP-class10'

top: 'precision-class10'

top: 'recall-class10'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class11'

bottom: 'bbox-list-label-class11'

bottom: 'bbox-list-class11'

top: 'bbox-list-scored-class11'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class11'

bottom: 'bbox-list-scored-class11'

top: 'mAP-class11'

top: 'precision-class11'

top: 'recall-class11'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class12'

bottom: 'bbox-list-label-class12'

bottom: 'bbox-list-class12'

top: 'bbox-list-scored-class12'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class12'

bottom: 'bbox-list-scored-class12'

top: 'mAP-class12'

top: 'precision-class12'

top: 'recall-class12'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'score-class13'

bottom: 'bbox-list-label-class13'

bottom: 'bbox-list-class13'

top: 'bbox-list-scored-class13'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'ScoreDetections'

}

include: { phase: TEST stage: "val" }

}

layer {

type: 'Python'

name: 'mAP-class13'

bottom: 'bbox-list-scored-class13'

top: 'mAP-class13'

top: 'precision-class13'

top: 'recall-class13'

python\_param {

module: 'caffe.layers.detectnet.mean\_ap'

layer: 'mAP'

param\_str : '1920, 1088, 16'

}

include: { phase: TEST stage: "val" }

}