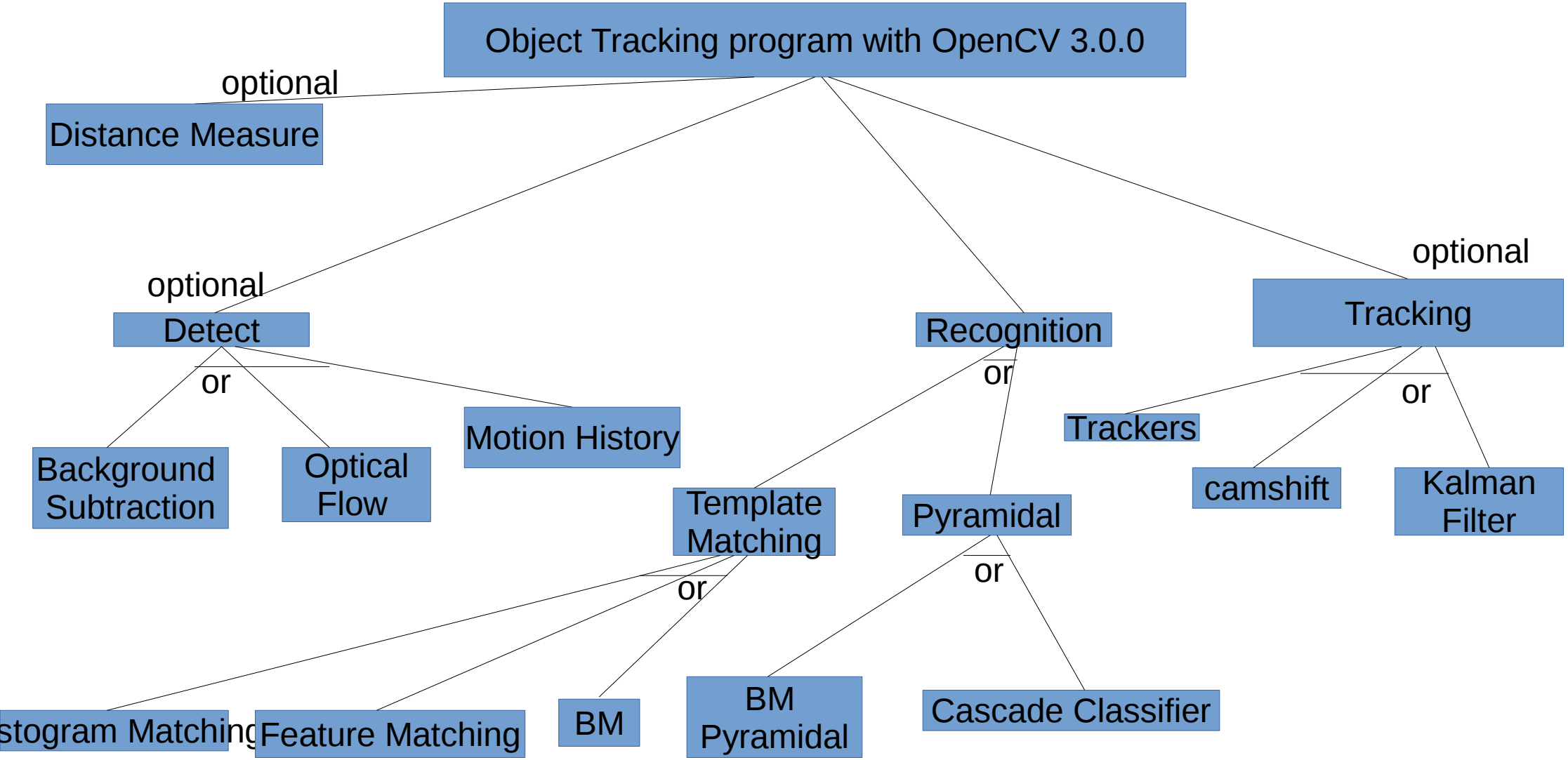
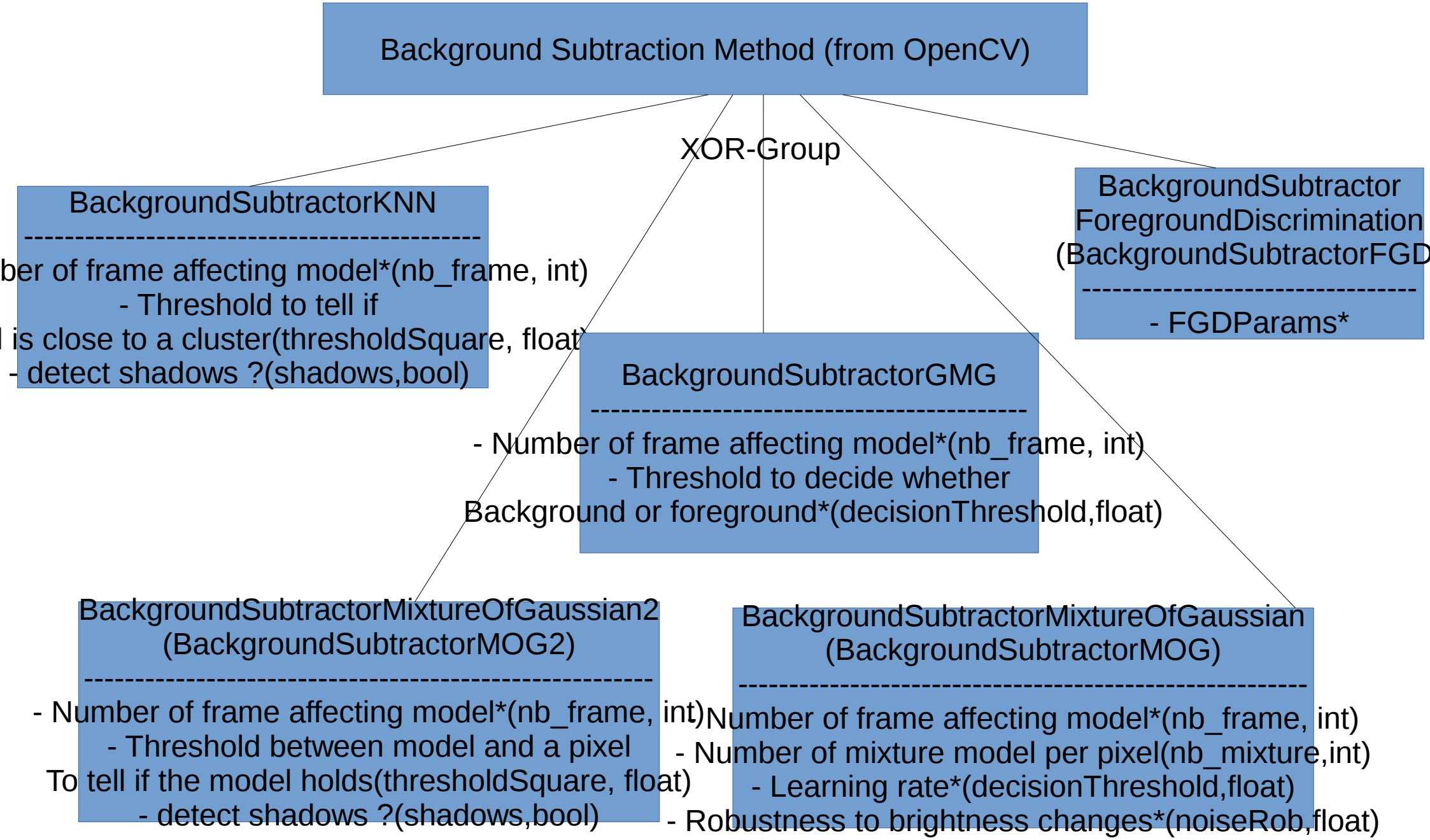
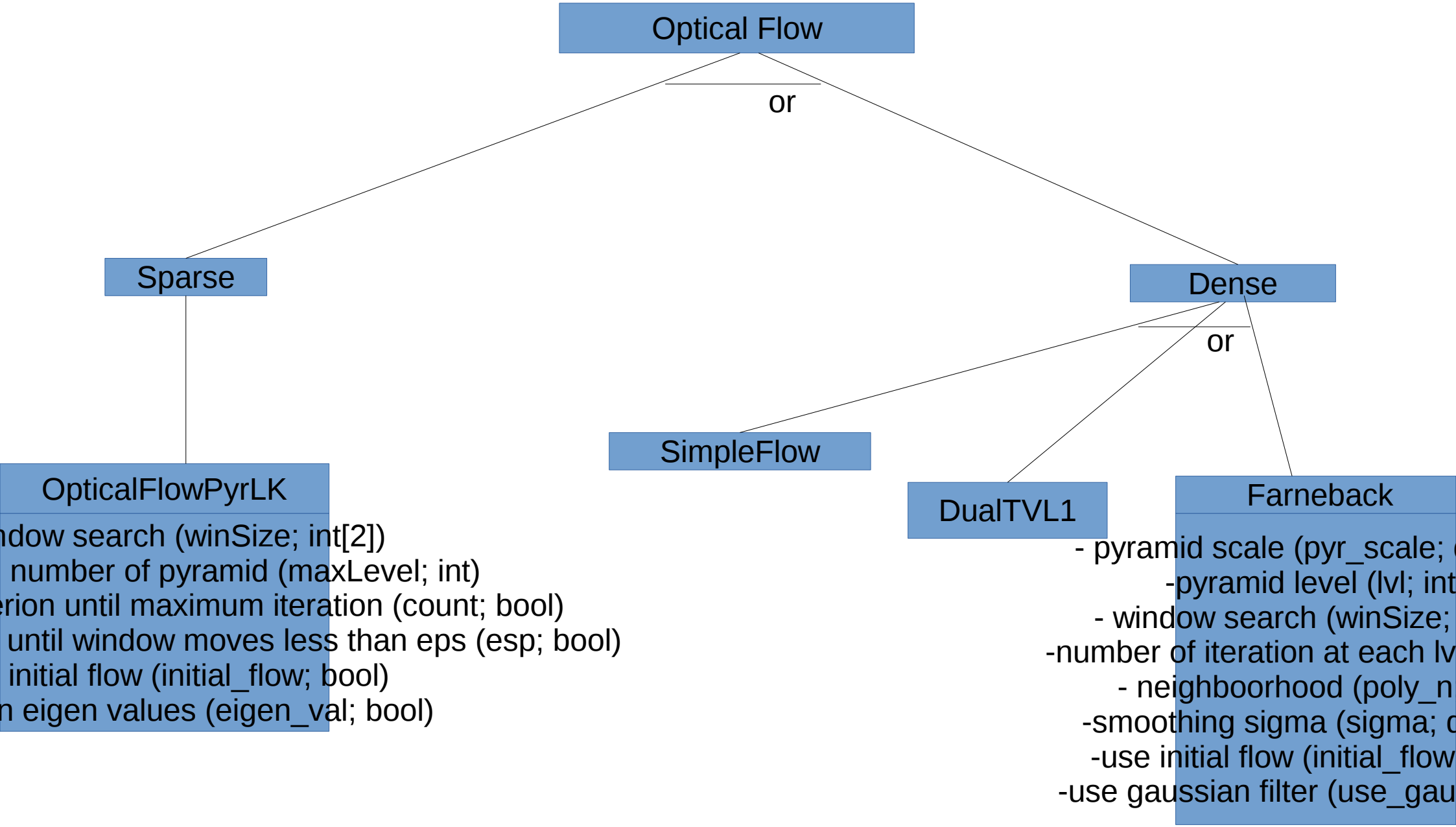
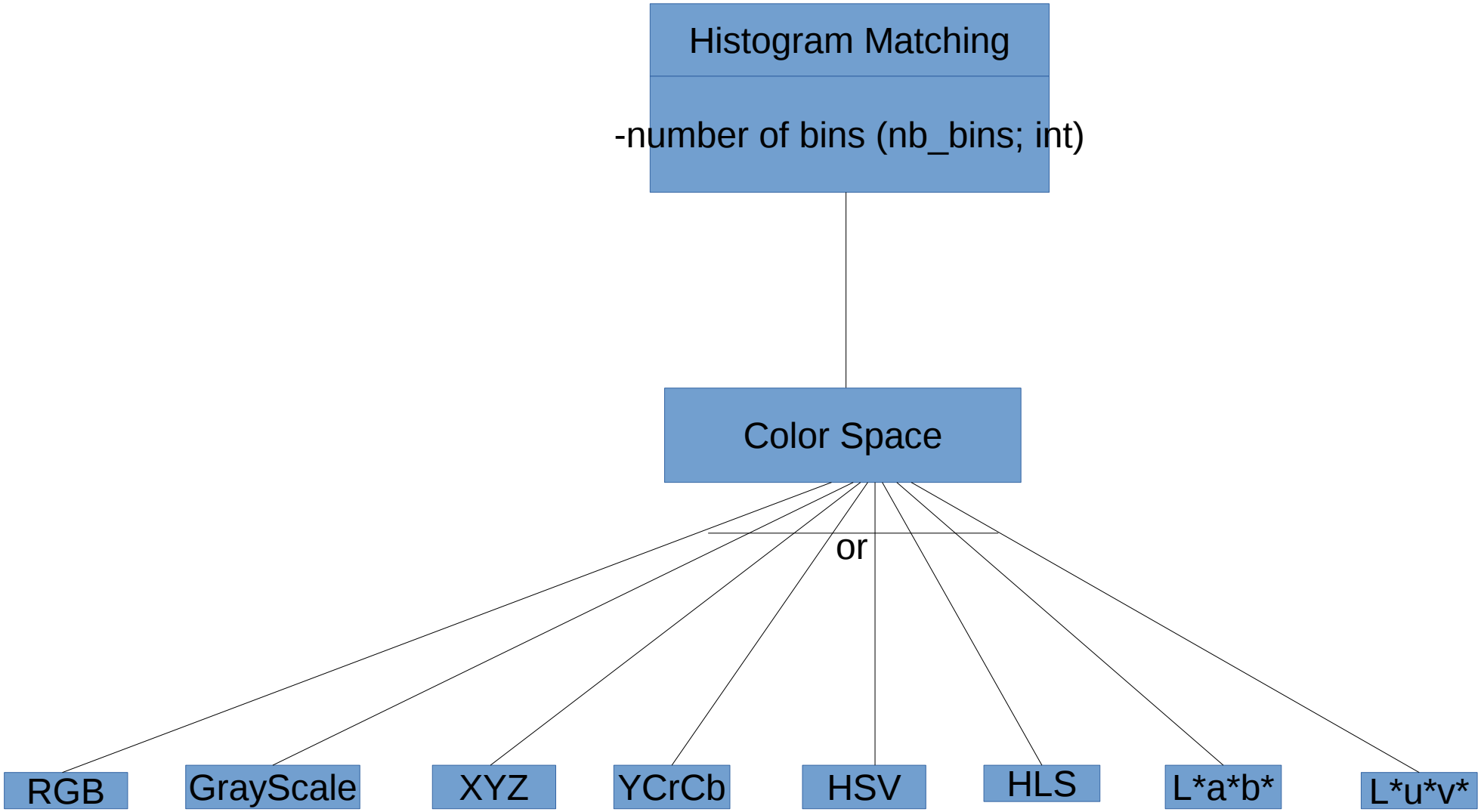


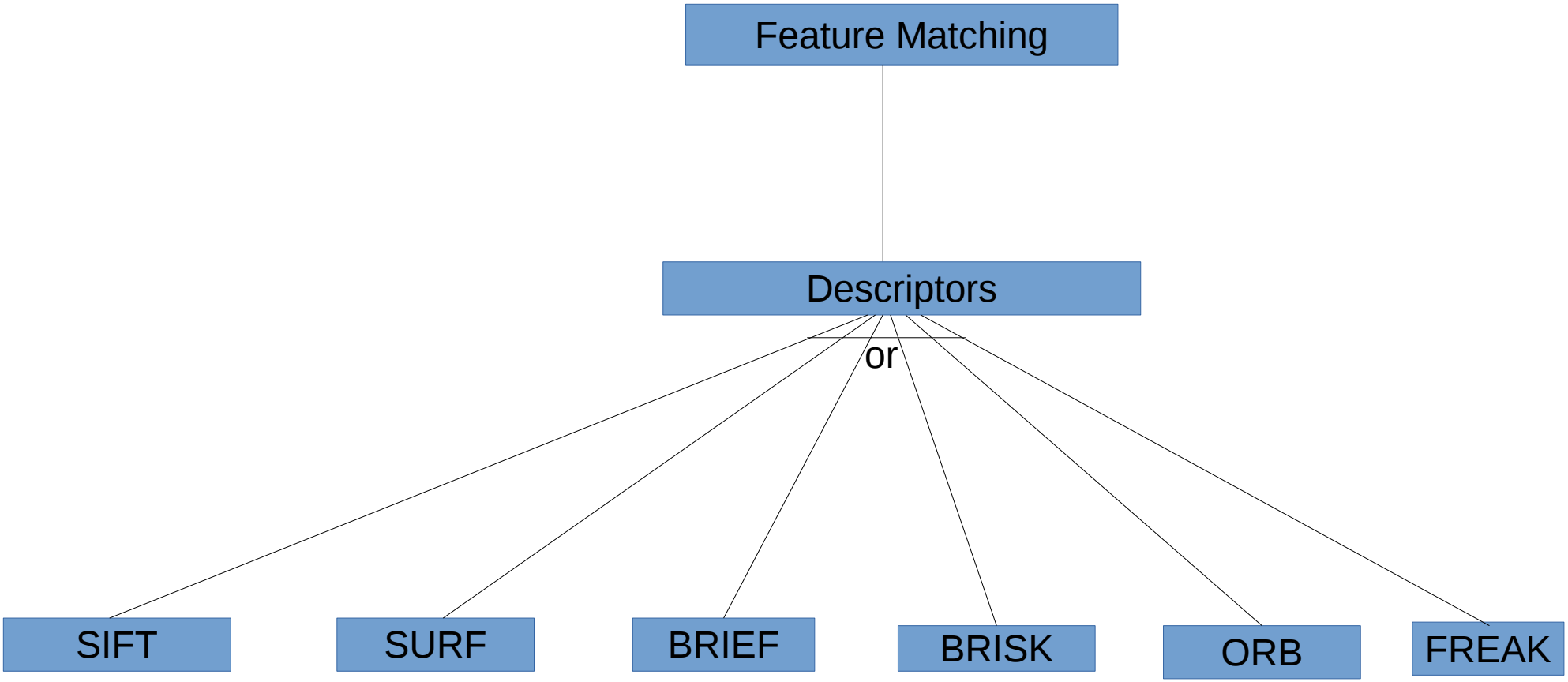
Histogram implies Detect
Feature matching implies Detect
BM implies Detect

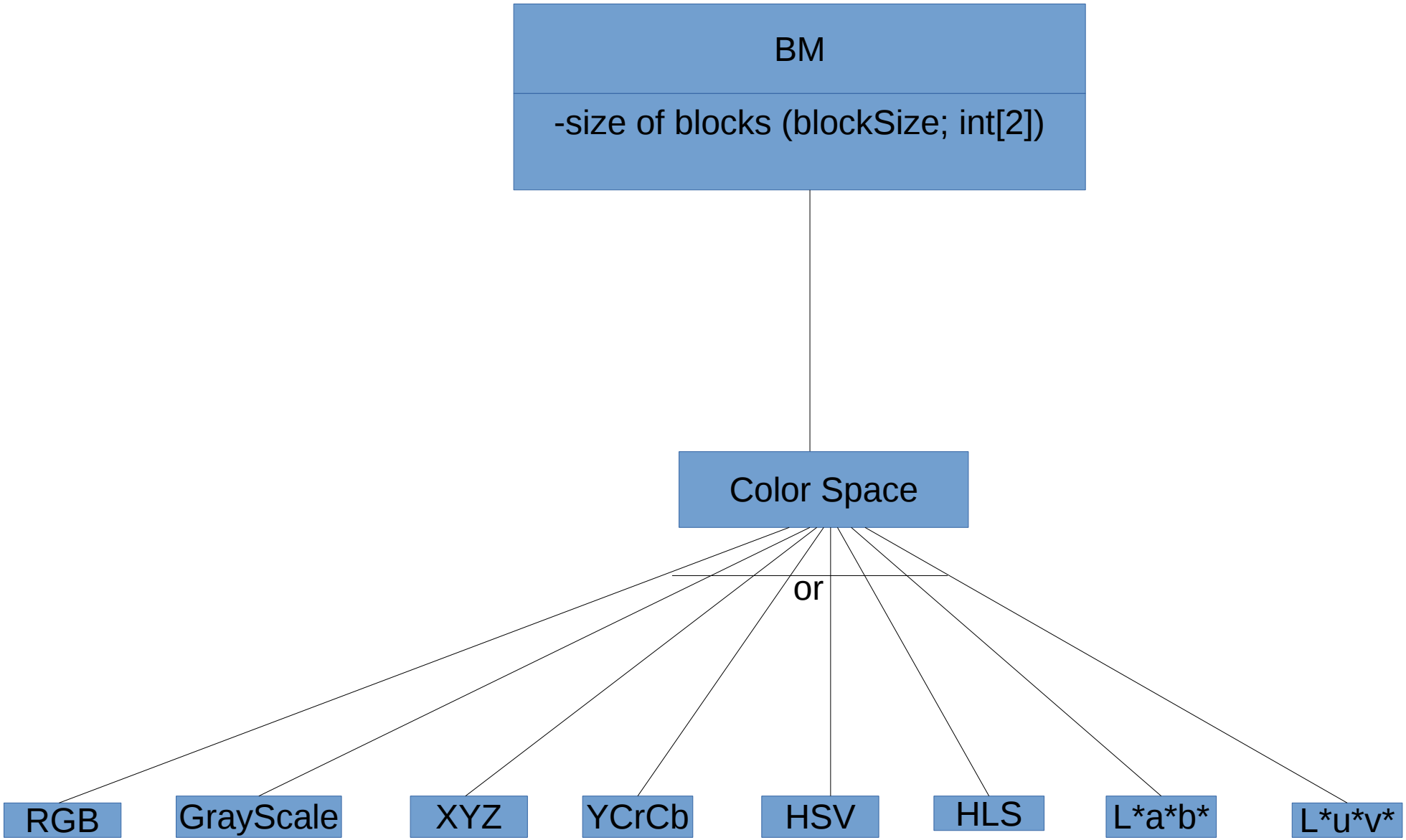
Histogram Matching implies Distance Measure
BM implies Distance Measure
BM Pyramidal implies Distance Measure

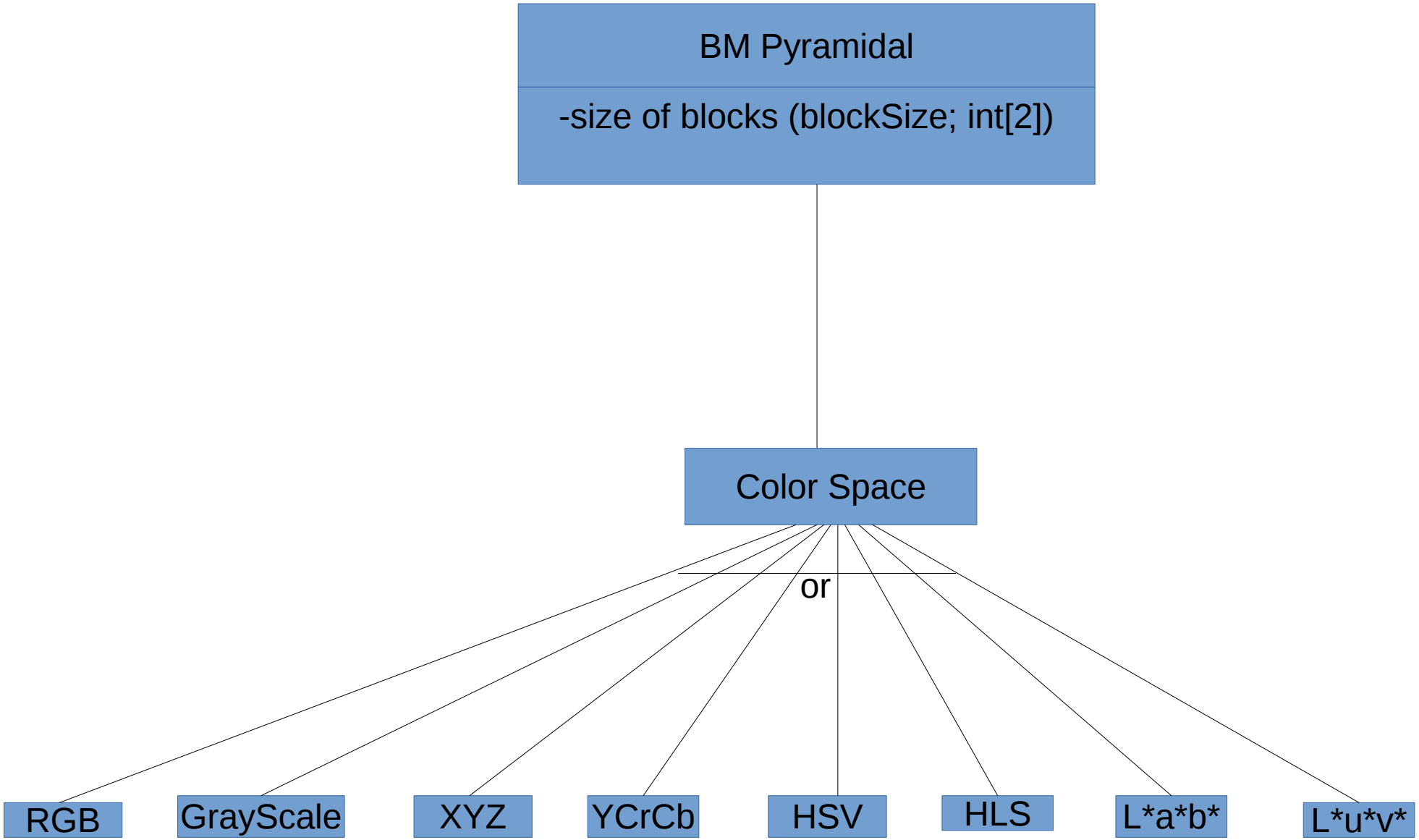

Background Subtraction Method (from OpenCV)
XOR-Group
BackgroundSubtractorKNN
ber of frame affecting model*(nb_frame, int)
- Threshold to tell if
l is close to a cluster(thresholdSquare, float)
- detect shadows ?(shadows,bool)
BackgroundSubtractor
ForegroundDiscrimination
(BackgroundSubtractorFGD
- FGDParams*
BackgroundSubtractorGMG
- Number of frame affecting model*(nb_frame, int)
- Threshold to decide whether
Background or foreground*(decisionThreshold,float)
BackgroundSubtractorMixtureOfGaussian2
(BackgroundSubtractorMOG2)
- Number of frame affecting model*(nb_frame, int)
- Threshold between model and a pixel
To tell if the model holds(thresholdSquare, float)
- detect shadows ?(shadows,bool)
BackgroundSubtractorMixtureOfGaussian
(BackgroundSubtractorMOG)
Number of frame affecting model*(nb_frame, int)
- Number of mixture model per pixel(nb_mixture,int)
- Learning rate*(decisionThreshold,float)
- Robustness to brightness changes*(noiseRob,float)

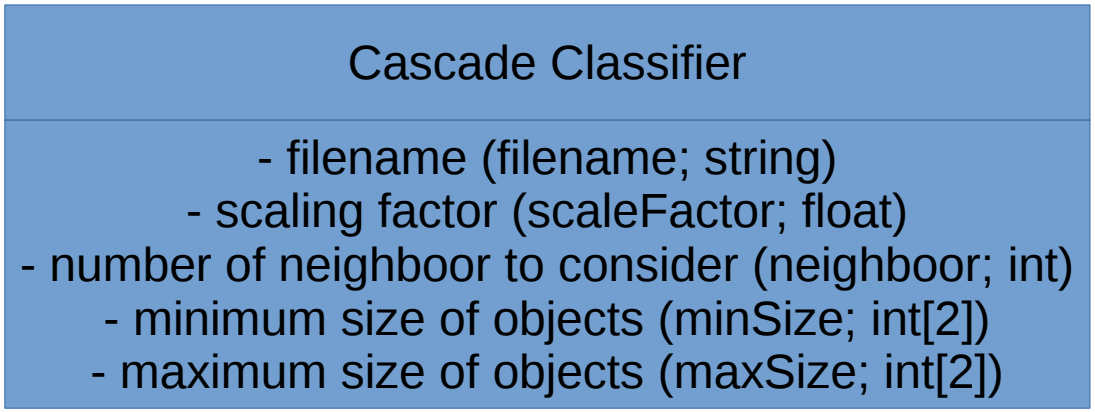

Optical Flow
or
Sparse
Dense
or
SimpleFlow
DualTVL1
Farneback
OpticalFlowPyrLK
ndow search (winSize; int[2])
number of pyramid (maxLevel; int)
erion until maximum iteration (count; bool)
until window moves less than eps (esp; bool)
initial flow (initial_flow; bool)
n eigen values (eigen_val; bool)
- pyramid scale (pyr_scale;
-pyramid level (lvl; int
- window search (winSize;
-number of iteration at each lv
- neighboorhood (poly_n
-smoothing sigma (sigma;
-use initial flow (initial_flow
-use gaussian filter (use_gau


Histogram Matching
-number of bins (nb_bins; int)
Color Space
or
RGB
GrayScale
XYZ
YCrCb
HSV
HLS
L*a*b*
L*u*v*


Feature Matching
Descriptors
or
SIFT
SURF
BRIEF
BRISK
ORB
FREAK

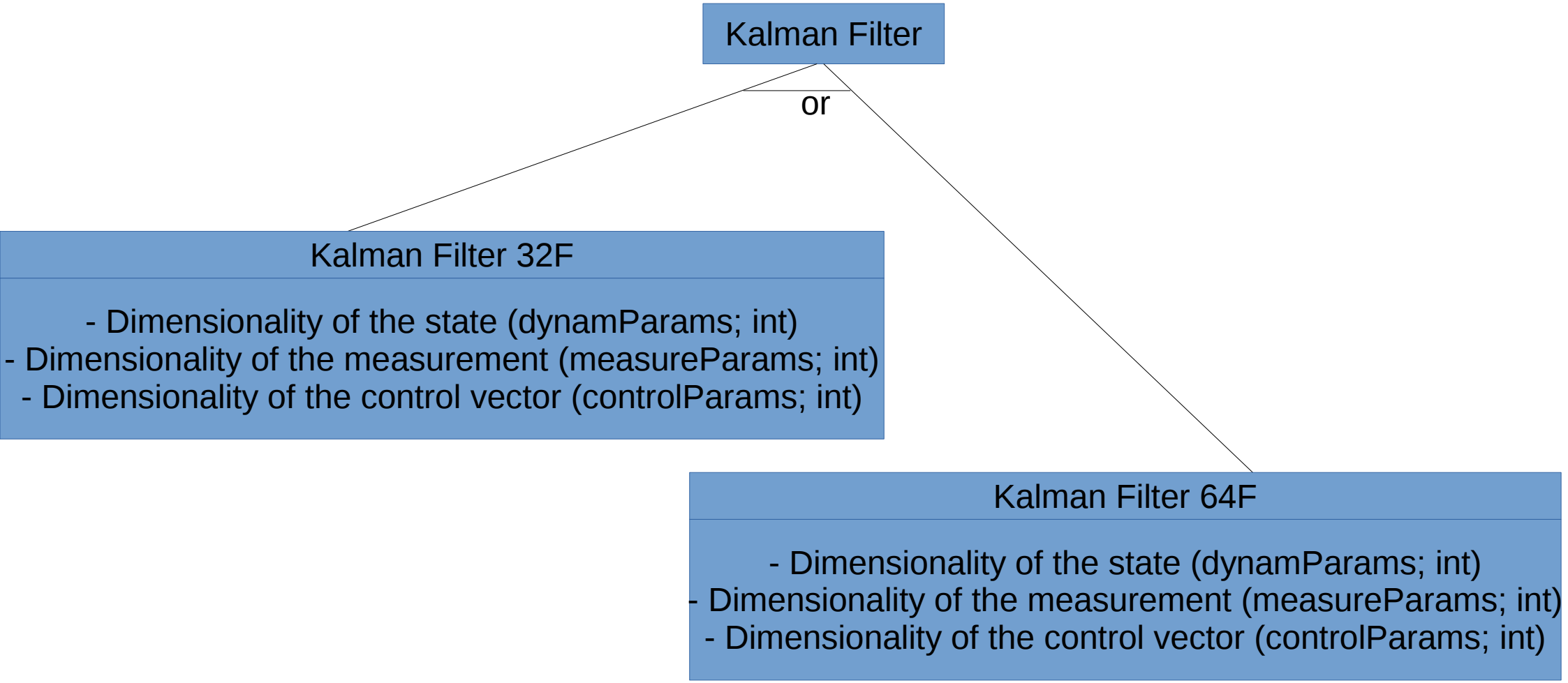

BM
-size of blocks (blockSize; int[2])
Color Space
or
RGB
GrayScale
XYZ
YCrCb
HSV
HLS
L*a*b*
L*u*v*

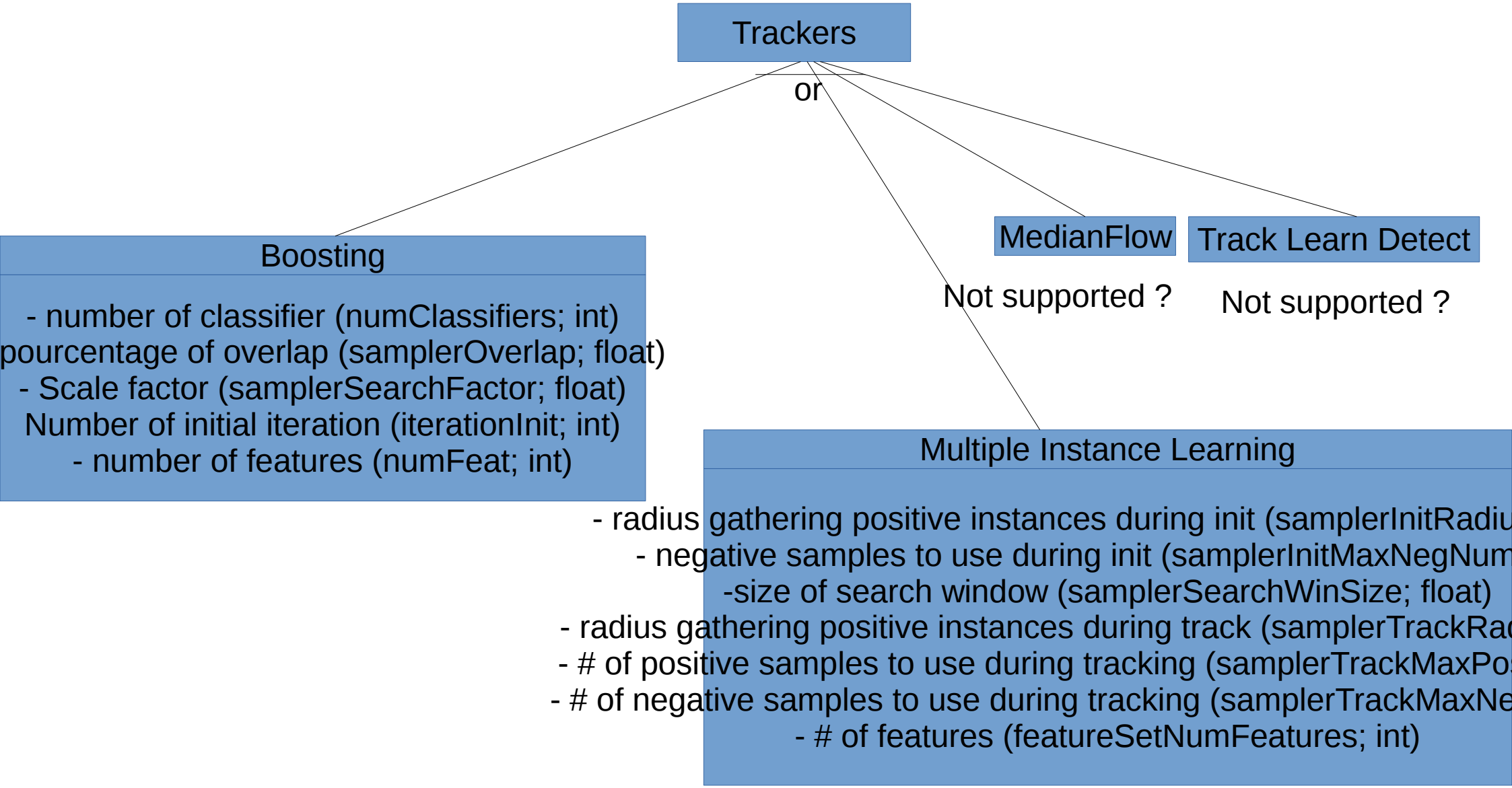

BM Pyramidal
-size of blocks (blockSize; int[2])
Color Space
or
RGB
GrayScale
XYZ
YCrCb
HSV
HLS
L*a*b*
L*u*v*


Cascade Classifier
- filename (filename; string)
- scaling factor (scaleFactor; float)
- number of neighboor to consider (neighboor; int)
- minimum size of objects (minSize; int[2])
- maximum size of objects (maxSize; int[2])

Camshift

?


Kalman Filter
or
Kalman Filter 32F
- Dimensionality of the state (dynamParams; int)
- Dimensionality of the measurement (measureParams; int)
- Dimensionality of the control vector (controlParams; int)
Kalman Filter 64F
- Dimensionality of the state (dynamParams; int)
- Dimensionality of the measurement (measureParams; int)
- Dimensionality of the control vector (controlParams; int)


Trackers
or
Boosting
- number of classifier (numClassifiers; int)
pourcentage of overlap (samplerOverlap; float)
- Scale factor (samplerSearchFactor; float)
Number of initial iteration (iterationInit; int)
- number of features (numFeat; int)
MedianFlow
Not supported ?
Track Learn Detect
Not supported ?
Multiple Instance Learning
- radius gathering positive instances during init (samplerInitRadiu
- negative samples to use during init (samplerInitMaxNegNum
-size of search window (samplerSearchWinSize; float)
- radius gathering positive instances during track (samplerTrackRa
- # of positive samples to use during tracking (samplerTrackMaxPo
- # of negative samples to use during tracking (samplerTrackMaxNe
- # of features (featureSetNumFeatures; int)