#BGK CNT

self.minPixelStabilityCNT = self.ui.SBMinPS.value()

self.maxPixelStabilityCNT = self.ui.SBMaxPS.value()

self.useHistoryCNT = self.ui.CBHistory.isChecked()

self.isParallelCNT = self.ui.CBIsParallel.isChecked()

#BGK GMG

self.initializationFramesGMG = self.ui.SBInitialF.value()

self.decisionThresholdGMG = self.ui.DSBThreshold.value()

#BGK GSOC

self.nSamplesGSOC = self.ui.SBNSamples.value()

self.replaceRateGSOC = self.ui.DSBReplace.value()

self.propagationRateGSOC = self.ui.DSBPropagation.value()

self.hitsThresholdGSOC = self.ui.SBNThreshold.value()

self.alphaGSOC = self.ui.DSBalpha.value()

self.betaGSOC = self.ui.DSBBeta.value()

#BGK KNN

self.historyKNN = self.ui.SBHistory.value()

self.dist2ThresholdKNN = self.ui.DSBThreshold\_2.value()

self.detectShadowsKNN = self.ui.CBDShadowsKNN.isChecked()

#BKG LSBP

self.nSamplesLSBP = self.ui.SBNSamplesLSBP.value()

self.LSBPRadius = self.ui.SBLRadius.value()

self.TupperLSBP = self.ui.DSBTupper.value()

self.RscaleLSBP = self.ui.DSRScale.value()

self.LSBPthreshold = self.ui.DSBLThreshold.value()

self.minCountLSBP = self.ui.DSBPmCount.value()

#BKG MOG

self.historyMOG = self.ui.SBHistoryMOG.value()

self.nmixturesMOG = self.ui.SBMixtures.value()

self.backgroundRatioMOG = self.ui.DSBRatio.value()

self.noiseSigmaMOG = self.ui.SBNoise.value()

#BKG MOG2

self.historyMOG2 = self.ui.SBHistoryMOG2.value()

self.varThresholdMOG2 = self.ui.SBThresholdMOG2.value()

self.detectShadowsMOG2 = self.ui.CBDshadowsMOG2.isChecked()

#BKG RNB

self.alphaRNB = self.ui.DSBAlpha.value()

self.ThuRNB = self.ui.SBThu.value()

self.ThsRNB = self.ui.SBThs.value()

#BKG FZBL

self.alphaFZBL = self.ui.DSBAlphaFL.value()

self.ThsFZBL = self.ui.SBThSFL.value()

self.ThfsFZBL= self.ui.DSBThfsFL.value()

#BKG FZB

self.alphaFZB = self.ui.DSBAlphaFZB.value()

self.alphaMinFZB = self.ui.DSBAlphaFZB\_2.value()

self.ThsFZB = self.ui.SBThSFZB.value()

self.ThfsFZB = self.ui.DSBThfsFZB.value()

#Image Img Subtractor

self.alphaSUB = self.ui.DSBAlphaI.value()

self.bethaSUB = self.ui.DSBBetaI.value()

self.constantSUB = self.ui.DSBConstant.value()

#Image Lapse

self.alphaLapse = self.ui.DSBAlphaL.value()

self.bethaLapse = self.ui.DSBBetaL.value()

self.constantLapse = self.ui.DSBConstantL.value()

#BGK CNT

self.minPixelStabilityCNT

self.maxPixelStabilityCNT

self.useHistoryCNT

self.isParallelCNT

#BGK GMG

self.initializationFramesGMG

self.decisionThresholdGMG

#BGK GSOC

self.nSamplesGSOC

self.replaceRateGSOC

self.propagationRateGSOC

self.hitsThresholdGSOC

self.alphaGSOC

self.betaGSOC

#BGK KNN

self.historyKNN

self.dist2ThresholdKNN

self.detectShadowsKNN

#BKG LSBP

self.nSamplesLSBP

self.LSBPRadius

self.TupperLSBP

self.RscaleLSBP

self.LSBPthreshold

self.minCountLSBP

#BKG MOG

self.historyMOG

self.nmixturesMOG

self.backgroundRatioMOG

self.noiseSigmaMOG

#BKG MOG2

self.historyMOG2

self.varThresholdMOG2

self.detectShadowsMOG2

#BKG RNB

self.alphaRNB

self.ThuRNB

self.ThsRNB

#BKG FZBL

self.alphaFZBL

self.ThsFZBL

self.ThfsFZBL

#BKG FZB

self.alphaFZB

self.alphaMinFZB

self.ThsFZB

self.ThfsFZB

#Image Img Subtractor

self.alphaSUB

self.bethaSUB

self.constantSUB

#Image Lapse

self.alphaLapse

self.bethaLapse

self.constantLapse

# BGK CNT  
'minPixelStabilityCNT': self.minPixelStabilityCNT,  
'maxPixelStabilityCNT': self.maxPixelStabilityCNT,  
'useHistoryCNT': self.useHistoryCNT,  
'isParallelCNT': self.isParallelCNT,  
# BGK GMG  
'initializationFramesGMG': self.initializationFramesGMG,  
'decisionThresholdGMG': self.decisionThresholdGMG,  
# BGK GSOC  
'nSamplesGSOC': self.nSamplesGSOC,  
'replaceRateGSOC': self.replaceRateGSOC,  
'propagationRateGSOC': self.propagationRateGSOC,  
'hitsThresholdGSOC': self.hitsThresholdGSOC,  
'alphaGSOC': self.alphaGSOC,  
'betaGSOC': self.betaGSOC,  
# BGK KNN  
'historyKNN': self.historyKNN,  
'dist2ThresholdKNN': self.dist2ThresholdKNN,  
'detectShadowsKNN': self.detectShadowsKNN,  
# BKG LSBP  
'nSamplesLSBP': self.nSamplesLSBP,  
'LSBPRadius': self.LSBPRadius,  
'TupperLSBP': self.TupperLSBP,  
'RscaleLSBP': self.RscaleLSBP,  
'LSBPthreshold': self.LSBPthreshold,  
'minCountLSBP': self.minCountLSBP,  
# BKG MOG  
'historyMOG': self.historyMOG,  
'nmixturesMOG': self.nmixturesMOG,  
'backgroundRatioMOG': self.backgroundRatioMOG,  
'noiseSigmaMOG': self.noiseSigmaMOG,  
# BKG MOG2  
'historyMOG2': self.historyMOG2,  
'varThresholdMOG2': self.varThresholdMOG2,  
'detectShadowsMOG2': self.detectShadowsMOG2,  
# BKG RNB  
'alphaRNB': self.alphaRNB,  
'ThuRNB': self.ThuRNB,  
'ThsRNB': self.ThsRNB,  
# BKG FZBL  
'alphaFZBL': self.alphaFZBL,  
'ThsFZBL': self.ThsFZBL,  
'ThfsFZBL': self.ThfsFZBL,  
# BKG FZB  
'alphaFZB': self.alphaFZB,  
'alphaMinFZB': self.alphaMinFZB,  
'ThsFZB': self.ThsFZB,  
'ThfsFZB': self.ThfsFZB,  
# Image Img Subtractor  
'alphaSUB': self.alphaSUB,  
'bethaSUB': self.bethaSUB,  
'constantSUB': self.constantSUB,  
# Image Lapse  
'alphaLapse': self.alphaLapse,  
'bethaLapse': self.bethaLapse,  
'constantLapse': self.constantLapse

Thu = 30  
#Ths = 30  
#Thfs = 0.4  
Ths = self.ThsFZBL  
Thfs = self.ThfsFZBL  
alphamin = 0.9

img = cv2.addWeighted(fmg, self.alphaSUB,  
 (np.where(bmg > fmg, img, 0)), self.bethaSUB, self.constantSUB)

Thu = 30  
#Ths = 30  
#Thfs = 0.4  
#alphamin = 0.9  
Ths = self.ThsFZB  
Thfs = self.ThfsFZB  
alphamin = self.alphaMinFZB

img = cv2.addWeighted(fmg, self.alphaSUB,  
 (np.where(bmg > fmg, img, 0)), self.bethaSUB, self.constantSUB)

#alpha = 0.9  
#Thu = 30  
#Ths = 30  
alpha = self.alphaRNB  
Thu = self.ThuRNB  
Ths = self.ThsRNB