

360AOIs

1.0

Generated by Doxygen 1.9.7

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 File Index	7
3.1 File List	7
4 Class Documentation	9
4.1 Animation Class Reference	9
4.1.1 Detailed Description	12
4.1.2 Member Function Documentation	12
4.1.2.1 Awake()	12
4.1.2.2 CalculateInterpolatedFrames()	12
4.1.2.3 CreateKeyframe()	13
4.1.2.4 DeleteKeyframeFromRightClickMenu()	14
4.1.2.5 GetKeyframeIndexByFrame()	14
4.1.2.6 HideKeyframes()	15
4.1.2.7 IsFrameInKeyframes()	15
4.1.2.8 IsObjectInCorrectState()	15
4.1.2.9 LoadKeyframes()	16
4.1.2.10 ObjectChangeStarted()	17
4.1.2.11 OnActivate()	17
4.1.2.12 OnDeactivate()	17
4.1.2.13 OnDestroy()	18
4.1.2.14 SetKeyframeVisibility()	18
4.1.2.15 SetObjectsStateToFrameData()	18
4.1.2.16 SetupRightClickMenu()	19
4.1.2.17 ShowKeyframes()	20
4.1.2.18 Start()	20
4.1.2.19 ToggleKeyframeVisibilityFromRightClickMenu()	21
4.1.2.20 Update()	21
4.1.2.21 UpdateKeyframe()	21
4.1.2.22 UpdateOrCreateKeyframeOnObjectChangeCompleted()	22
4.1.3 Member Data Documentation	23
4.1.3.1 _movementComponent	23
4.1.3.2 _objectIsBeingChanged	23
4.1.3.3 _resizeComponent	23
4.1.3.4 KeyframePrefab	23
4.1.4 Property Documentation	23
4.1.4.1 InterpolatedFrames	23
4.1.4.2 Keyframes	23

4.2 AOIComponent Class Reference	24
4.2.1 Detailed Description	25
4.2.2 Member Function Documentation	25
4.2.2.1 ActiveAOIchanged()	25
4.2.2.2 OnActivate()	25
4.2.2.3 OnDeactivate()	26
4.2.2.4 OnDestroy()	26
4.2.2.5 Start()	27
4.3 AOIManager Class Reference	27
4.3.1 Detailed Description	29
4.3.2 Member Function Documentation	29
4.3.2.1 Awake()	29
4.3.2.2 CompleteManualResizingDueToOverlap()	30
4.3.2.3 OnTestSceneToggled()	30
4.3.2.4 OnVrStatusChange()	31
4.3.2.5 SetActiveObject()	31
4.3.2.6 SpawnAoi()	32
4.3.2.7 SpawnAOIatCursorPosition()	33
4.3.2.8 Start()	33
4.3.2.9 StartManualResizingDueToOverlap()	33
4.3.2.10 ToggleAOIColliders()	34
4.3.2.11 Update()	34
4.3.3 Member Data Documentation	35
4.3.3.1 _canvas	35
4.3.3.2 _manuallyTriggeredMovementDueToOverlap	35
4.3.3.3 _manuallyTriggeredResizeDueToOverlap	35
4.3.3.4 _objectToManuallyResize	35
4.3.3.5 ActiveAOI	35
4.3.3.6 AOIPrefab	35
4.3.3.7 FlipMesh	35
4.3.3.8 Instance	36
4.3.3.9 NewActiveAOI	36
4.3.3.10 SpawnDistance	36
4.4 EventDetection.AoiParameters Class Reference	36
4.4.1 Constructor & Destructor Documentation	36
4.4.1.1 AoiParameters()	36
4.4.2 Member Data Documentation	37
4.4.2.1 AoiName	37
4.4.2.2 DurationOfFirstFixation	37
4.4.2.3 DwellCount	37
4.4.2.4 EntryTime	37
4.4.2.5 FirstPassDwellTime	37

4.4.2.6 FixationCount	37
4.4.2.7 ProportionOfFixations	37
4.4.2.8 ProportionOfTotalDwellTime	37
4.4.2.9 TimeToFirstFixation	37
4.4.2.10 TotalDwellTime	38
4.5 SaveAndLoadAOIs.AOI_SaveData Class Reference	38
4.5.1 Member Data Documentation	38
4.5.1.1 Keyframes	38
4.5.1.2 Name	38
4.6 CameraDrag Class Reference	38
4.6.1 Detailed Description	39
4.6.2 Member Function Documentation	39
4.6.2.1 Start()	39
4.6.2.2 Update()	40
4.6.3 Member Data Documentation	40
4.6.3.1 _canvas	40
4.6.3.2 _X	40
4.6.3.3 _Y	40
4.6.3.4 Speed	40
4.7 ChangeColorOnDwell Class Reference	41
4.7.1 Detailed Description	42
4.7.2 Member Function Documentation	42
4.7.2.1 OnRaycastEnter()	42
4.7.2.2 OnRaycastExit()	43
4.7.2.3 Start()	43
4.7.3 Member Data Documentation	43
4.7.3.1 _renderer	43
4.7.3.2 RaycastEnterColor	43
4.7.3.3 RaycastExitColor	43
4.8 ClickableOnTimeline Class Reference	43
4.8.1 Detailed Description	45
4.8.2 Member Function Documentation	45
4.8.2.1 OnPointerDown()	45
4.8.2.2 OnPointerEnter()	46
4.8.2.3 OnPointerExit()	46
4.8.2.4 OnPointerUp()	47
4.8.2.5 Start()	47
4.8.2.6 Update()	47
4.8.3 Member Data Documentation	48
4.8.3.1 _rect	48
4.8.4 Property Documentation	48
4.8.4.1 IsClickedAndMouseStillDown	48

4.8.4.2 IsHovered	48
4.8.4.3 Timeline	48
4.9 Colorable Class Reference	49
4.9.1 Detailed Description	50
4.9.2 Member Enumeration Documentation	50
4.9.2.1 AoiColors	50
4.9.3 Member Function Documentation	51
4.9.3.1 OnActivate()	51
4.9.3.2 OnDeactivate()	51
4.9.3.3 SetColor() [1/2]	51
4.9.3.4 SetColor() [2/2]	52
4.9.3.5 Start()	53
4.9.4 Member Data Documentation	53
4.9.4.1 _color	53
4.9.4.2 _renderer	53
4.10 SaveAndLoadAOIs.Datalist Class Reference	53
4.10.1 Member Data Documentation	54
4.10.1.1 list	54
4.11 DebugReRun Class Reference	54
4.11.1 Member Function Documentation	55
4.11.1.1 CSVtoDataSampleList()	55
4.11.1.2 DebugFixationSampleFlagsToCsv()	55
4.11.1.3 OnReRun()	56
4.12 DisplayAsFrameOnTimeline Class Reference	56
4.12.1 Detailed Description	57
4.12.2 Member Function Documentation	58
4.12.2.1 Hide()	58
4.12.2.2 RecalculateFrameBasedOnPosition()	58
4.12.2.3 Show()	58
4.12.2.4 Start()	59
4.12.2.5 Update()	59
4.12.3 Member Data Documentation	59
4.12.3.1 Frame	59
4.12.4 Property Documentation	59
4.12.4.1 CanvasGroup	59
4.12.4.2 IsDraggable	59
4.12.4.3 Rect	60
4.12.4.4 Timeline	60
4.13 DraggableAlongXYPlane Class Reference	60
4.13.1 Detailed Description	61
4.13.2 Member Function Documentation	61
4.13.2.1 Move()	61

4.13.2.2 OnDisable()	62
4.13.2.3 OnMouseDown()	62
4.13.2.4 OnMouseUp()	62
4.13.2.5 Update()	62
4.13.3 Member Data Documentation	63
4.13.3.1 _offset	63
4.13.3.2 FlipMesh	63
4.13.3.3 MovementCompleted	63
4.13.3.4 MovementStarted	63
4.13.4 Property Documentation	63
4.13.4.1 IsTryingToMove	63
4.14 DraggableAroundCamera Class Reference	63
4.14.1 Detailed Description	65
4.14.2 Member Function Documentation	65
4.14.2.1 Move()	65
4.14.2.2 OnActivate()	66
4.14.2.3 OnDeactivate()	66
4.14.2.4 OnMouseDown()	66
4.14.2.5 OnMouseUp()	67
4.14.2.6 Update()	67
4.14.3 Member Data Documentation	67
4.14.3.1 _offset	67
4.14.3.2 DistanceToCamera	67
4.14.3.3 MovementCompleted	67
4.14.3.4 MovementStarted	68
4.14.4 Property Documentation	68
4.14.4.1 IsTryingToMove	68
4.14.4.2 IsTryingToMoveDueToBeingActivted	68
4.15 DraggableOnTimeline Class Reference	68
4.15.1 Detailed Description	70
4.15.2 Member Function Documentation	70
4.15.2.1 MovementCompleted()	70
4.15.2.2 MovementCondition()	71
4.15.2.3 OnMovementCompleted()	71
4.15.2.4 OnPointerDown()	71
4.15.2.5 OnPositionChange()	72
4.15.2.6 SetAnchoredX()	72
4.15.2.7 Start()	72
4.15.2.8 Update()	72
4.15.3 Member Data Documentation	73
4.15.3.1 _mouseOffset	73
4.15.3.2 MovementCompletedSubscribers	73

4.15.4 Property Documentation	73
4.15.4.1 Canvas	73
4.15.4.2 IsBeingDragged	73
4.16 EditorCamera Class Reference	73
4.16.1 Detailed Description	74
4.16.2 Member Function Documentation	74
4.16.2.1 OnRecordingToggled()	74
4.16.2.2 OnTestSceneToggled()	75
4.16.2.3 OnVrStatusChange()	75
4.16.2.4 Start()	76
4.16.2.5 Update()	76
4.16.3 Member Data Documentation	76
4.16.3.1 Camera	76
4.16.3.2 EditorCamera_GameObject	77
4.17 EventDetection.Event Class Reference	77
4.17.1 Member Enumeration Documentation	77
4.17.1.1 EventType	77
4.17.2 Constructor & Destructor Documentation	77
4.17.2.1 Event()	77
4.17.3 Member Data Documentation	78
4.17.3.1 AoiName	78
4.17.3.2 Duration	78
4.17.3.3 EndTimestamp	78
4.17.3.4 StartTimestamp	78
4.17.3.5 Type	78
4.18 EventDetection Class Reference	78
4.18.1 Detailed Description	80
4.18.2 Constructor & Destructor Documentation	80
4.18.2.1 EventDetection()	80
4.18.3 Member Function Documentation	81
4.18.3.1 DetectDwells()	81
4.18.3.2 DetectFixations()	81
4.18.3.3 EventCountProportion()	82
4.18.3.4 EventDurationProportion()	83
4.18.3.5 FilterInvalidData()	83
4.18.3.6 FindIndexOfNextValidSample()	84
4.18.3.7 GetSampleIndexAtMinFixationDuration()	84
4.18.3.8 UniqueAoiNames()	85
4.18.4 Member Data Documentation	85
4.18.4.1 DwellToleranceInMs	85
4.18.4.2 FilteredDataSamples	85
4.18.4.3 FixationDispersionThresholdInDegrees	85

4.18.4.4 FixationDurationThresholdInMs	86
4.18.4.5 Fixationsample_flags	86
4.18.4.6 FixationTimeBetweenSamplesLimitInMs	86
4.18.4.7 FixationVelocityThresholdInDegPerSecond	86
4.18.4.8 RawDataSamples	86
4.18.4.9 Results	86
4.18.5 Property Documentation	86
4.18.5.1 Dwells	86
4.18.5.2 Fixations	86
4.19 EventDetectionTester Class Reference	87
4.19.1 Detailed Description	88
4.19.2 Member Function Documentation	88
4.19.2.1 RunTest()	88
4.19.2.2 Start()	88
4.20 EyeRecorder Class Reference	88
4.20.1 Detailed Description	90
4.20.2 Member Function Documentation	90
4.20.2.1 EyeCallback()	90
4.20.2.2 FinishRecording()	91
4.20.2.3 OnApplicationQuit()	91
4.20.2.4 OnDisable()	92
4.20.2.5 Release()	92
4.20.2.6 SetIsRecordingAndNotify()	93
4.20.2.7 Start()	93
4.20.2.8 StartRecording()	93
4.20.2.9 ToggleRecording()	94
4.20.2.10 Update()	94
4.20.3 Member Data Documentation	94
4.20.3.1 _eye_callback_registered	94
4.20.3.2 _recordedEyeData	95
4.20.3.3 _timestampOffset	95
4.20.3.4 Recording_ID	95
4.20.3.5 Sensitivity	95
4.20.4 Property Documentation	95
4.20.4.1 CurrentEyeData	95
4.20.4.2 Instance	95
4.20.4.3 IsRecording	95
4.20.4.4 OnRecordingToggled	95
4.20.4.5 PreviousEyeData	95
4.20.4.6 RawData	96
4.21 EyeTrackingDataSample Class Reference	96
4.21.1 Detailed Description	97

4.21.2 Constructor & Destructor Documentation	97
4.21.2.1 EyeTrackingDataSample() [1/2]	97
4.21.2.2 EyeTrackingDataSample() [2/2]	97
4.21.3 Member Data Documentation	97
4.21.3.1 AoiName	97
4.21.3.2 GazeDirection	97
4.21.3.3 Timestamp	97
4.21.3.4 VerboseEyeData	97
4.21.3.5 VideoFrame	98
4.22 FPSCounter Class Reference	98
4.22.1 Detailed Description	99
4.22.2 Member Function Documentation	99
4.22.2.1 Awake()	99
4.22.2.2 Update()	99
4.22.3 Member Data Documentation	99
4.22.3.1 _averageCounter	99
4.22.3.2 _averageFromAmount	99
4.22.3.3 _cacheNumbersAmount	99
4.22.3.4 _currentAveraged	100
4.22.3.5 _frameRateSamples	100
4.22.3.6 CachedNumberStrings	100
4.22.3.7 Text	100
4.23 Animation.FrameData Class Reference	100
4.23.1 Member Data Documentation	100
4.23.1.1 Position	100
4.23.1.2 Rotation	100
4.23.1.3 Ulrepresentation	100
4.23.1.4 Vertices	101
4.23.1.5 VideoFrame	101
4.23.1.6 Visibility	101
4.24 GazeRaycaster Class Reference	101
4.24.1 Detailed Description	102
4.24.2 Member Function Documentation	102
4.24.2.1 Awake()	102
4.24.2.2 GazeRaycast()	103
4.24.2.3 OnNothingHit()	103
4.24.2.4 OnObjectHit()	103
4.24.2.5 ToggleGazeRaycastingVrStatusChange()	104
4.24.3 Member Data Documentation	104
4.24.3.1 Instance	104
4.24.4 Property Documentation	105
4.24.4.1 CurrentlyHitObject	105

4.25 GazeRayRenderer Class Reference	105
4.25.1 Detailed Description	106
4.25.2 Member Function Documentation	106
4.25.2.1 Awake()	106
4.25.2.2 RenderCurrentGazeAsRay()	107
4.25.2.3 ToggleGazeRenderingOnVrStatusChange()	107
4.25.3 Member Data Documentation	108
4.25.3.1 _GazeRayRenderer	108
4.25.3.2 _gazeSphere	108
4.25.3.3 _gazeSphereRenderer	108
4.25.3.4 _LengthOfRay	108
4.25.3.5 _ray_color	108
4.25.3.6 Instance	108
4.26 GazeSimulator Class Reference	109
4.26.1 Detailed Description	110
4.26.2 Member Function Documentation	110
4.26.2.1 GenerateFixation()	110
4.26.2.2 GetNextFakeSample()	111
4.26.2.3 OnDrawGizmos()	111
4.26.2.4 Start()	111
4.26.3 Member Data Documentation	111
4.26.3.1 fake_data	111
4.26.3.2 fake_time_in_ms	112
4.26.3.3 TestRayDirection	112
4.26.3.4 UseSimulatedDataInstead	112
4.27 GazeToTextureMapper Class Reference	112
4.27.1 Detailed Description	113
4.27.2 Member Function Documentation	113
4.27.2.1 OnDrawGizmos()	113
4.27.2.2 VectorToUV()	113
4.27.3 Member Data Documentation	114
4.27.3.1 direction	114
4.27.3.2 texelX	114
4.27.3.3 texelY	114
4.27.3.4 VideoHeight	114
4.27.3.5 VideoWidth	114
4.28 HandleForResizable Class Reference	114
4.28.1 Detailed Description	115
4.28.2 Member Function Documentation	116
4.28.2.1 OnMouseDown()	116
4.28.2.2 OnMouseUp()	116
4.28.2.3 Start()	116

4.28.3 Member Data Documentation	116
4.28.3.1 Parent	116
4.29 Helper Class Reference	117
4.29.1 Member Function Documentation	117
4.29.1.1 MostCommon< T >()	117
4.30 Listable Class Reference	117
4.30.1 Detailed Description	120
4.30.2 Member Function Documentation	120
4.30.2.1 OnActivate()	120
4.30.2.2 OnDeactivate()	120
4.30.2.3 OnDestroy()	120
4.30.2.4 OnListItemClick()	120
4.30.2.5 operator string()	121
4.30.2.6 SetName()	121
4.30.2.7 Start()	121
4.30.2.8 UpdateNameFromInputField()	122
4.30.3 Member Data Documentation	122
4.30.3.1 _contentContainer	122
4.30.3.2 _inputField	122
4.30.3.3 _itemInstance	123
4.30.3.4 ItemPrefab	123
4.30.3.5 ListableCount	123
4.30.4 Property Documentation	123
4.30.4.1 Name	123
4.31 LogMessagesManager Class Reference	123
4.31.1 Detailed Description	124
4.31.2 Member Function Documentation	125
4.31.2.1 Awake()	125
4.31.2.2 CloseLogPanel()	125
4.31.2.3 OnLogMessageReceived()	125
4.31.2.4 OpenLogPanel()	126
4.31.2.5 Update()	126
4.31.3 Member Data Documentation	126
4.31.3.1 ContentContainer	126
4.31.3.2 LogEntryPrefab	126
4.31.3.3 LogPanel	126
4.31.3.4 ScrollRect	127
4.31.4 Property Documentation	127
4.31.4.1 Instance	127
4.32 MenuCloseLogic Class Reference	127
4.32.1 Detailed Description	128
4.32.2 Member Function Documentation	128

4.32.2.1 Update()	128
4.33 Playhead Class Reference	128
4.33.1 Detailed Description	130
4.33.2 Member Function Documentation	130
4.33.2.1 MovementCondition()	130
4.33.2.2 OnMovementCompleted()	131
4.33.2.3 OnPositionChange()	131
4.33.2.4 SetAnchoredX()	131
4.33.2.5 Update()	132
4.33.3 Member Data Documentation	132
4.33.3.1 CurrentFrameLabel	132
4.33.3.2 CurrentFrameLabelWithBG	132
4.34 PreVideoTestScene Class Reference	133
4.34.1 Detailed Description	133
4.34.2 Member Function Documentation	134
4.34.2.1 Awake()	134
4.34.2.2 OnPreVideoTestSceneToggled()	134
4.34.2.3 OnRecordingToggled()	134
4.34.2.4 OnVrStatusChange()	135
4.35 RaycastHitHandler Class Reference	136
4.35.1 Detailed Description	137
4.35.2 Member Function Documentation	137
4.35.2.1 OnRaycastEnter()	137
4.35.2.2 OnRaycastExit()	137
4.35.3 Property Documentation	137
4.35.3.1 IsCurrentlyHitByRaycast	137
4.36 RecordButton Class Reference	138
4.36.1 Detailed Description	139
4.36.2 Member Function Documentation	139
4.36.2.1 SetImageOnRecordingToggle()	139
4.36.2.2 SetInteractivityBasedOnVrStatus()	139
4.36.2.3 Start()	140
4.36.3 Member Data Documentation	140
4.36.3.1 _offSprite	140
4.36.3.2 _onSprite	140
4.36.3.3 _recButton	140
4.36.3.4 _recButtonImage	141
4.37 Resizable Class Reference	141
4.37.1 Detailed Description	143
4.37.2 Member Function Documentation	143
4.37.2.1 HideHandles()	143
4.37.2.2 OnActivate()	144

4.37.2.3 OnDeactivate()	144
4.37.2.4 OnDrawGizmos()	144
4.37.2.5 RecenterOrigin()	145
4.37.2.6 RecenterOriginBasedOnKeyframeData()	145
4.37.2.7 Resize()	146
4.37.2.8 SetIsTryingToResize()	146
4.37.2.9 ShowHandles()	147
4.37.2.10 Start()	147
4.37.2.11 Update()	148
4.37.3 Member Data Documentation	148
4.37.3.1 _collider	148
4.37.3.2 _handles	148
4.37.3.3 _renderer	148
4.37.3.4 _vertices	148
4.37.3.5 HandlePrefab	148
4.37.3.6 ResizingCompleted	148
4.37.3.7 ResizingStarted	148
4.37.4 Property Documentation	149
4.37.4.1 IsTryingToResize	149
4.37.4.2 Mesh	149
4.38 RightClickMenu Class Reference	149
4.38.1 Detailed Description	151
4.38.2 Member Function Documentation	151
4.38.2.1 AddItem()	151
4.38.2.2 Awake()	151
4.38.2.3 DeleteObjectUsingRightClickMenu()	152
4.38.2.4 GetChildGameObject()	152
4.38.2.5 GotClicked()	152
4.38.2.6 HideMenu()	153
4.38.2.7 IsCursorOverUI()	153
4.38.2.8 OnDestroy()	154
4.38.2.9 RemoveItem()	154
4.38.2.10 ShowMenu()	154
4.38.2.11 Update()	155
4.38.3 Member Data Documentation	155
4.38.3.1 _contentContainer	155
4.38.3.2 _itemPrefab	155
4.38.3.3 _menuPrefab	155
4.38.3.4 AddDeleteOnStart	155
4.38.3.5 canvas	155
4.38.3.6 LastClickedObjectWithMenu	155
4.38.4 Property Documentation	156

4.38.4.1 Menu	156
4.39 SaveAndLoadAOIs Class Reference	156
4.39.1 Detailed Description	157
4.39.2 Member Function Documentation	157
4.39.2.1 LoadAoIs()	157
4.39.2.2 OnLoadAOIButtonDown()	158
4.39.2.3 Save()	158
4.39.3 Member Data Documentation	158
4.39.3.1 <i>datalist</i>	158
4.40 SequenceChartPythonProcess Class Reference	159
4.40.1 Detailed Description	159
4.40.2 Member Function Documentation	159
4.40.2.1 CallSequenceChartScript()	159
4.41 Setup Class Reference	160
4.41.1 Detailed Description	161
4.41.2 Member Function Documentation	161
4.41.2.1 SetFramerateOnVREnable()	161
4.41.2.2 Start()	161
4.42 EventDetectionTester.TesterSample Class Reference	162
4.42.1 Member Enumeration Documentation	164
4.42.1.1 <i>aoi_test_selector</i>	164
4.42.2 Constructor & Destructor Documentation	164
4.42.2.1 TesterSample()	164
4.42.3 Member Data Documentation	164
4.42.3.1 <i>fake_time</i>	164
4.42.3.2 <i>ms_per_videoframe</i>	164
4.43 Timeline Class Reference	164
4.43.1 Detailed Description	167
4.43.2 Member Function Documentation	167
4.43.2.1 Awake()	167
4.43.2.2 ChangeTimelineScale()	167
4.43.2.3 FixPlayheadOvershoot()	168
4.43.2.4 HoverHandoverWithChildren()	168
4.43.2.5 OnNewFrameInVideoPlayer()	169
4.43.2.6 OnPointerEnter()	170
4.43.2.7 OnPointerExit()	170
4.43.2.8 OnRecordingToggled()	170
4.43.2.9 OnRectTransformDimensionsChange()	171
4.43.2.10 OnVideoManagerInitialized()	172
4.43.2.11 OnVideoPlayerNewFrameCount()	172
4.43.2.12 SetBlockEdgeSnapping()	173
4.43.2.13 SetEndFrame()	174

4.43.2.14 SetStartFrame()	174
4.43.2.15 SetTargetFrame() [1/2]	175
4.43.2.16 SetTargetFrame() [2/2]	175
4.43.2.17 SetVideoStartFrame()	176
4.43.2.18 Start()	177
4.43.2.19 SyncTimelineWithVideo()	177
4.43.2.20 Update()	178
4.43.3 Member Data Documentation	178
4.43.3.1 _playhead	178
4.43.3.2 EndFrameLabel	178
4.43.3.3 OnTargetFrameChange	179
4.43.3.4 PlayheadRect	179
4.43.3.5 StartFrameLabel	179
4.43.4 Property Documentation	179
4.43.4.1 BlockedEdgeSnapping	179
4.43.4.2 EndFrame	179
4.43.4.3 Height	179
4.43.4.4 Instance	179
4.43.4.5 IsHovered	179
4.43.4.6 ScrollScale	179
4.43.4.7 StartFrame	179
4.43.4.8 TargetFrame	180
4.43.4.9 VideoStartFrame	180
4.43.4.10 Width	180
4.43.4.11 WidthPerFrame	180
4.43.4.12 ZoomFactor	180
4.44 VideoManager Class Reference	180
4.44.1 Detailed Description	182
4.44.2 Member Function Documentation	182
4.44.2.1 Awake()	182
4.44.2.2 OnNewFrameInTimeline()	183
4.44.2.3 OnNewFrameReady()	183
4.44.2.4 OnRecordingToggled()	184
4.44.2.5 OnVideoLoaded()	184
4.44.2.6 OnVideoPlayerError()	185
4.44.2.7 OnVideoPlayerPrepareCompleted()	185
4.44.2.8 OnVolumeSliderValueChanged()	186
4.44.2.9 PlayPause()	186
4.44.2.10 PlayUntilFrameOne()	187
4.44.2.11 SeekCompleted()	187
4.44.2.12 SeekFrameInVideo()	187
4.44.2.13 Start()	188

4.44.2.14 Update()	188
4.44.3 Member Data Documentation	189
4.44.3.1 _isSeeking	189
4.44.3.2 Instance	189
4.44.3.3 SkyboxMaterial	189
4.44.3.4 TargetTexture	189
4.44.3.5 VolumeSlider	189
4.44.4 Property Documentation	189
4.44.4.1 CurrentFrame	189
4.44.4.2 FrameCount	189
4.44.4.3 IsInitialized	189
4.44.4.4 OnFrameCountChange	189
4.44.4.5 OnNewFrame	190
4.44.4.6 OnVideoManagerInitialized	190
4.44.4.7 VideoPlayer	190
4.45 VisualAngleTester Class Reference	190
4.45.1 Member Function Documentation	192
4.45.1.1 OnMovementCompleted()	192
4.45.1.2 Start()	192
4.45.1.3 Update()	192
4.45.2 Member Data Documentation	192
4.45.2.1 _angle_sphere	192
4.45.2.2 _meshRenderer	192
4.45.2.3 _movementComponent	193
4.45.2.4 Color	193
4.45.2.5 distance	193
4.45.2.6 phi	193
4.45.2.7 size	193
4.45.2.8 TestDirection	193
4.46 VRButton Class Reference	193
4.46.1 Detailed Description	194
4.46.2 Member Function Documentation	195
4.46.2.1 Start()	195
4.46.2.2 UpdateImageBasedOnVrManagerStatus()	195
4.46.2.3 UpdateInteractivityBasedOnRecordingStatus()	195
4.46.3 Member Data Documentation	196
4.46.3.1 _errorSprite	196
4.46.3.2 _loadingSprite	196
4.46.3.3 _offSprite	196
4.46.3.4 _onSprite	196
4.46.3.5 _vrButton	196
4.46.3.6 _vrButtonImage	196

4.47 VRCamera Class Reference	197
4.47.1 Detailed Description	198
4.47.2 Member Function Documentation	198
4.47.2.1 Awake()	198
4.47.2.2 OnAOIVisibilityToggled()	198
4.47.2.3 OnPreVideoStimulusToggled()	199
4.47.2.4 OnRecordingToggled()	199
4.47.2.5 OnVRStatusChange()	200
4.48 VRManager Class Reference	200
4.48.1 Detailed Description	202
4.48.2 Member Enumeration Documentation	202
4.48.2.1 VRStatus	202
4.48.3 Member Function Documentation	202
4.48.3.1 SetCurrentVRStatusAndNotify()	202
4.48.3.2 StartXR()	203
4.48.3.3 StartXRCoroutine()	203
4.48.3.4 StopXR()	204
4.48.3.5 ToggleXR()	205
4.48.4 Member Data Documentation	205
4.48.4.1 _EyeFramework	205
4.48.4.2 _XRRig	205
4.48.5 Property Documentation	205
4.48.5.1 CurrentVRStatus	205
4.48.5.2 OnXRStatusUpdate	205
4.49 VrSettingsMenu Class Reference	206
4.49.1 Detailed Description	207
4.49.2 Member Function Documentation	208
4.49.2.1 LaunchCalibration()	208
4.49.2.2 OnNewDwellToleranceEntered()	208
4.49.2.3 OnNewFixationDispersionThresholdEntered()	208
4.49.2.4 OnNewFixationDurationThresholdEntered()	208
4.49.2.5 OnNewFixationVelocityThresholdEntered()	209
4.49.2.6 Start()	209
4.49.2.7 ToggleAOIVisibilityInVr()	209
4.49.2.8 ToggleGazeVisibilityInVr()	209
4.49.2.9 TogglePreVideoTestScene()	209
4.49.2.10 ToggleVisibility()	209
4.49.2.11 ToggleVRCameraTracking()	210
4.49.2.12 Update()	210
4.49.3 Member Data Documentation	210
4.49.3.1 AOIVisibilityToggle	210
4.49.3.2 AreAOIsVisibleInVr	210

4.49.3.3 CameraTrackingToggle	210
4.49.3.4 DwellToleranceInput	210
4.49.3.5 EyeFramework	210
4.49.3.6 FixationDispersionThresholdInput	211
4.49.3.7 FixationDurationThresholdInput	211
4.49.3.8 FixationVelocityThresholdInput	211
4.49.3.9 GazeVisibilityToggle	211
4.49.3.10 IsGazeVisibleInVr	211
4.49.3.11 IsPreVideoTestSceneEnabled	211
4.49.3.12 IsVRCameraTracked	211
4.49.3.13 menu	211
4.49.3.14 OnAOIVisibilityToggled	211
4.49.3.15 OnIsPreVideoTestSceneEnabledToggled	211
4.49.3.16 PreVideoTestSceneToggle	211
5 File Documentation	213
5.1 Assets/Scripts/AOI Management/AOI Components/Animation.cs File Reference	213
5.2 Assets/Scripts/AOI Management/AOI Components/AOIComponent.cs File Reference	213
5.3 Assets/Scripts/AOI Management/AOI Components/Colorable.cs File Reference	213
5.4 Assets/Scripts/AOI Management/AOI Components/DraggableAroundCamera.cs File Reference	214
5.5 Assets/Scripts/AOI Management/AOI Components>Listable.cs File Reference	214
5.6 Assets/Scripts/AOI Management/AOI Components/Resizable.cs File Reference	214
5.7 Assets/Scripts/AOI Management/AOI Handles/DraggableAlongXYPlane.cs File Reference	214
5.8 Assets/Scripts/AOI Management/AOI Handles/HandleForResizable.cs File Reference	214
5.9 Assets/Scripts/AOI Management/AOIManager.cs File Reference	215
5.10 Assets/Scripts/AOI Management/CameraDrag.cs File Reference	215
5.11 Assets/Scripts/AOI Management/EditorCamera.cs File Reference	215
5.12 Assets/Scripts/Application/FPSCounter.cs File Reference	215
5.13 Assets/Scripts/Application/LogMessagesManager.cs File Reference	215
5.14 Assets/Scripts/Application/MenuCloseLogic.cs File Reference	216
5.15 Assets/Scripts/Application/RightClickMenu.cs File Reference	216
5.16 Assets/Scripts/Application/Setup.cs File Reference	216
5.17 Assets/Scripts/Application/VisualAngleTester.cs File Reference	216
5.18 Assets/Scripts/Eyetracking and Gaze Rendering/ChangeColorOnDwell.cs File Reference	216
5.19 Assets/Scripts/Eyetracking and Gaze Rendering/DebugReRun.cs File Reference	216
5.20 Assets/Scripts/Eyetracking and Gaze Rendering/EventDetection.cs File Reference	217
5.21 Assets/Scripts/Eyetracking and Gaze Rendering/EyeRecorder.cs File Reference	217
5.22 Assets/Scripts/Eyetracking and Gaze Rendering/EyeTrackingDataSample.cs File Reference	217
5.23 Assets/Scripts/Eyetracking and Gaze Rendering/GazeRaycaster.cs File Reference	217
5.24 Assets/Scripts/Eyetracking and Gaze Rendering/GazeRayRenderer.cs File Reference	218
5.25 Assets/Scripts/Eyetracking and Gaze Rendering/RaycastHitHandler.cs File Reference	218
5.26 Assets/Scripts/Eyetracking and Gaze Rendering/RecordButton.cs File Reference	218

5.27 Assets/Scripts/Eyetracking and Gaze Rendering/Simulation/EventDetectionTester.cs File Reference	218
5.28 Assets/Scripts/Eyetracking and Gaze Rendering/Simulation/GazeSimulator.cs File Reference	218
5.29 Assets/Scripts/Logging and Visualization/EyeDataLogger.cs File Reference	219
5.30 Assets/Scripts/Logging and Visualization/GazeToTextureMapper.cs File Reference	219
5.31 Assets/Scripts/Logging and Visualization/SequenceChartPythonProcess.cs File Reference	219
5.32 Assets/Scripts/Project Management/ProjectManager.cs File Reference	219
5.33 Assets/Scripts/Project Management/SaveAndLoadAOIs.cs File Reference	219
5.34 Assets/Scripts/Video and Timeline/ClickableOnTimeline.cs File Reference	219
5.35 Assets/Scripts/Video and Timeline/DisplayAsFrameOnTimeline.cs File Reference	220
5.36 Assets/Scripts/Video and Timeline/DraggableOnTimeline.cs File Reference	220
5.37 Assets/Scripts/Video and Timeline/Playhead.cs File Reference	220
5.38 Assets/Scripts/Video and Timeline/Timeline.cs File Reference	220
5.39 Assets/Scripts/Video and Timeline/VideoLoader.cs File Reference	220
5.40 Assets/Scripts/Video and Timeline/VideoManager.cs File Reference	220
5.41 Assets/Scripts/VR Management/PreVideoTestScene.cs File Reference	221
5.42 Assets/Scripts/VR Management/VRButton.cs File Reference	221
5.43 Assets/Scripts/VR Management/VRCamera.cs File Reference	221
5.44 Assets/Scripts/VR Management/VRManager.cs File Reference	221
5.45 Assets/Scripts/VR Management/VrSettingsMenu.cs File Reference	221
Index	223

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

EventDetection.AoiParameters	36
SaveAndLoadAOIs.AOISaveData	38
SaveAndLoadAOIs.Datalist	53
EventDetection.Event	77
EventDetection	78
EyeTrackingDataSample	96
EventDetectionTester.TesterSample	162
Animation.FrameData	100
Helper	117
IPointerDownHandler	
ClickableOnTimeline	43
DraggableOnTimeline	68
Playhead	128
IPointerEnterHandler	
ClickableOnTimeline	43
Timeline	164
IPointerExitHandler	
ClickableOnTimeline	43
Timeline	164
IPointerUpHandler	
ClickableOnTimeline	43
MonoBehaviour	
AOIComponent	24
Animation	9
Colorable	49
DraggableAroundCamera	63
Listable	117
Resizable	141
AOIManager	27
CameraDrag	38
ClickableOnTimeline	43
DebugReRun	54
DisplayAsFrameOnTimeline	56
DraggableAlongXYPlane	60
EditorCamera	73

EventDetectionTester	87
EyeRecorder	88
FPSCounter	98
GazeRayRenderer	105
GazeRaycaster	101
GazeSimulator	109
GazeToTextureMapper	112
HandleForResizable	114
LogMessagesManager	123
MenuCloseLogic	127
PreVideoTestScene	133
RaycastHitHandler	136
ChangeColorOnDwell	41
RecordButton	138
RightClickMenu	149
SaveAndLoadAOIs	156
SequenceChartPythonProcess	159
Setup	160
Timeline	164
VRButton	193
VRCamera	197
VRManager	200
VideoManager	180
VisualAngleTester	190
VrSettingsMenu	206

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

[Animation](#)

This script does a lot of things and should probably be refactored into multiple classes. a) Creates & Updates Keyframes for position, size and shape of AOI Does this by listening to DraggableAroundCameras and Resizables events b) Calculates position, size and shape for Frames between Keyframes using interpolation c) Animates object based on current frame of video by setting position, size and shape according to calculated frames 9

[AOIComponent](#)

This is the base class for components that make up an area of interest. The user can highlight and work with one AOI at a time, marking that one as active (done by the AOI Manager) This class subscribes to the event of changing the active AOI and specifies the relevant methods (OnActivate, OnDeactivate) that need to be implemented by derived classes 24

[AOIManager](#)

This class does 3 things 1.Manages AOIs based on user input a) Deactivates currently active AOIs and spawns new AOIs if user clicks on empty space in 360 degree video b) Sets an AOI as active when user clicks on AOI in 360 degree video and deactivates the previously selected AOI 27

[EventDetection.AoiParameters](#)

36

[SaveAndLoadAOIs.AOISaveData](#)

38

[CameraDrag](#)

This script is attached to the Editor Camera and allows the User to change the portion of the 360 degree video being displayed, by holding down right click and dragging. The camera then rotates based on the cursor movement 38

[ChangeColorOnDwell](#)

This script inherits from the [RaycastHitHandler](#) and therefore its functions are called by the [GazeRaycaster](#). It simply changes the color of the object it is attached to when the gaze of the person wearing the vr headset is over the object 41

[ClickableOnTimeline](#)

This class serves as a base for all clickable children of the timeline. it makes sure that the hover state between the timeline and its children is properly handed-over. For example if the user clicks on a keyframe and then drags the cursor, that should not move the playhead on the timeline because the user wanted to interact with the keyframe, not the timeline. Vice versa if the user scrubs the playhead across the timeline hovering over a keyframe should not trigger a click on the keyframe 43

[Colorable](#)

Class that was supposed to allow AOIs to be recolored from AOI List or [RightClickMenu](#). Not implemented though, so class is mostly useless. Currently only handles color of AOI based on whether or not AOI is selected by user 49

<code>SaveAndLoadAOIs.Datalist</code>	53
<code>DebugReRun</code>	54
<code>DisplayAsFrameOnTimeline</code>	
This classed is used for any UI element that should be displayed as a certain frame on the timeline for example a keyframe for frame 47 always needs to be positioned where frame 47 is on the timeline, regardless of scale / zoom level of the timeline. This class handles that	56
<code>DraggableAlongXYPlane</code>	
This script is attached to every AOI handle and allows the handle to be dragged along the XY plane of its parent(the AOI)	60
<code>DraggableAroundCamera</code>	
This script is attached to AOIs and allows them to be dragged around to position them within the 360 degree video	63
<code>DraggableOnTimeline</code>	
This class allows an object which is displayed on the timeline (using <code>DisplayAsFrameOnTimelineCComponent</code>) to be dragged along the timeline using the mouse. (Used by the playhead to scrub along the timeline)	68
<code>EditorCamera</code>	
This script is attached to the editor camera and does two things a) gives a static reference to the <code>EditorCamera</code> to be used for e.g. position calculation etc. b) matches the editor camera rotation to the VR cameras rotation if flag is set by user	73
<code>EventDetection.Event</code>	77
<code>EventDetection</code>	
This script contains all the event-detection algorithms for detecting dwells and fixations with their respective parameters like duration etc	78
<code>EventDetectionTester</code>	
Some testing code for the event detection. Generates a list <code>EyeTrackingDataSamples</code> , then creates an event detection object and passes the list as parameter. The created csv file can be checked manually and compared to the created <code>EyeTrackingDataSamples</code>	87
<code>EyeRecorder</code>	
Derived from SRanipal example script. This script is responsible for getting the eyedata from the VR-Headset, storing the data during recording as well as carrying out the raycast to check whether or not the gaze is hitting a AOI. Note: Callback runs on a separate thread to report at ~120hz. Unity is not threadsafe and cannot call any UnityEngine api from within callback thread	88
<code>EyeTrackingDataSample</code>	
Dataclass for storing all relevant data for a single eyetracking sample	96
<code>FPSCounter</code>	
Script to display FPS counter on the top right of screen. Source: https://forum.unity.com/threads/fps-counter.505495/	98
<code>Animation.FrameData</code>	100
<code>GazeRaycaster</code>	
Creates a raycast from current eyedata gaze direction. If an object with the <code>RaycastHitHandler</code> component is hit by racast this script calls the <code>OnRaycastEnter</code> and <code>OnRaycastExit</code> methods on entry and exit of the object. This allows objects to react to beeing looked at by the participant	101
<code>GazeRayRenderer</code>	
Derived from SRanipal Example Scripts. This script displays the current gaze of the participant as a colored ray within the experimentor-view of the application	105
<code>GazeSimulator</code>	
Another test-script. The code generates fixations at a specified Vector3. Instead of creating a eventdetection object here the <code>fake_data</code> is used in the eyerecorder class when the <code>UseSimulatedDataInstead</code> flag is set to true	109
<code>GazeToTextureMapper</code>	
Proof of concept class. Derived from shader code of Unity PanoramicShader. Allows to map the gaze direction onto pixel coordinates of the skybox-texture. Can be used in the future to create different types of visualizations	112
<code>HandleForResizable</code>	
This script is attached to every AOI-Handle. It signals the parent object that a handle has been clicked and resizing should start. Requires the parent to have the "Resizable" Component	114
<code>Helper</code>	117

Listable	This script manages the name of the AOI and its representation in the AOIList in the UI	117
LogMessagesManager	Script managing the creation of logs in the error/debug log console. Console can be opened and closed by pressing 'F1'	123
MenuCloseLogic	Tiny little script that's attached to everything that should be "closed" when the user clicks outside of it::s bounds. Mostly used for menus that should disappear when user clicks outside of them .	127
Playhead	Implements the playhead. Inherits DraggableOnTimeline	128
PreVideoTestScene	Handles display of testscene after vr is enabled but before recording starts	133
RaycastHitHandler	Abstract class that defines OnRaycastEnter and OnRaycastExit for interaction with GazeRaycaster . Should probably be an interface in the future	136
RecordButton	Script attached to record button at the top right corner of UI. Handles sprite changes based on events	138
Resizable	This script is attached to AOIs and is responsible for the creation of handles as well as resizing. Resizing happens when the user drags a handle of the currently selected AOI across the screen. Something of note: Resizing is done by changing the vertices of the AOIs mesh. Moving the vertices doesn't automatically recalculate the center (origin) of the game object, so this class manually recenters the objects origin during and after resizing is completed	141
RightClickMenu	This script can be attached to any gameobject to display a menu when the gameobject is right-clicked. Currently used on AOIs and keyframe-icons	149
SaveAndLoadAOIs	Handles saving and loading AOIs to and from a json file	156
SequenceChartPythonProcess	Calls python process that creates sequence chart from filteredData csv file	159
Setup	Currently doesn't do too much. Sets target framerate of application to 120Hz	160
EventDetectionTester.TesterSample	162
Timeline	Implements functionality of timeline. Allows timeline to be scrubbed and to zoom in on the timeline to be able to move playhead with more precision to better place keyframes	164
VideoManager	This class handles the videoplayer component and its events	180
VisualAngleTester	190
VRButton	Attached to VR Button. Manages interactability as well as sprite changes for button	193
VRCamera	Attached to the camera used by the VR headset and sets the cameras culling mask and clear flags	197
VRManager	Handles activation and deactivation of Unitys XR Framework and SRanipal Eye Framework	200
VrSettingsMenu	Attached to UI GameObject "TopBar". Handles all user input related to the VR Settings Menu and invokes events to notify other Scripts of setting-changes	206

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Assets/Scripts/AOI Management/ AOIManager.cs	215
Assets/Scripts/AOI Management/ CameraDrag.cs	215
Assets/Scripts/AOI Management/ EditorCamera.cs	215
Assets/Scripts/AOI Management/AOI Components/ Animation.cs	213
Assets/Scripts/AOI Management/AOI Components/ AOIComponent.cs	213
Assets/Scripts/AOI Management/AOI Components/ Colorable.cs	213
Assets/Scripts/AOI Management/AOI Components/ DraggableAroundCamera.cs	214
Assets/Scripts/AOI Management/AOI Components/ Listable.cs	214
Assets/Scripts/AOI Management/AOI Components/ Resizable.cs	214
Assets/Scripts/AOI Management/AOI Handles/ DraggableAlongXYPlane.cs	214
Assets/Scripts/AOI Management/AOI Handles/ HandleForResizable.cs	214
Assets/Scripts/Application/ FPSCounter.cs	215
Assets/Scripts/Application/ LogMessagesManager.cs	215
Assets/Scripts/Application/ MenuCloseLogic.cs	216
Assets/Scripts/Application/ RightClickMenu.cs	216
Assets/Scripts/Application/ Setup.cs	216
Assets/Scripts/Application/ VisualAngleTester.cs	216
Assets/Scripts/Eyetracking and Gaze Rendering/ ChangeColorOnDwell.cs	216
Assets/Scripts/Eyetracking and Gaze Rendering/ DebugReRun.cs	216
Assets/Scripts/Eyetracking and Gaze Rendering/ EventDetection.cs	217
Assets/Scripts/Eyetracking and Gaze Rendering/ EyeRecorder.cs	217
Assets/Scripts/Eyetracking and Gaze Rendering/ EyeTrackingDataSample.cs	217
Assets/Scripts/Eyetracking and Gaze Rendering/ GazeRaycaster.cs	217
Assets/Scripts/Eyetracking and Gaze Rendering/ GazeRayRenderer.cs	218
Assets/Scripts/Eyetracking and Gaze Rendering/ RaycastHitHandler.cs	218
Assets/Scripts/Eyetracking and Gaze Rendering/ RecordButton.cs	218
Assets/Scripts/Eyetracking and Gaze Rendering/Simulation/ EventDetectionTester.cs	218
Assets/Scripts/Eyetracking and Gaze Rendering/Simulation/ GazeSimulator.cs	218
Assets/Scripts/Logging and Visualization/ EyeDataLogger.cs	219
Assets/Scripts/Logging and Visualization/ GazeToTextureMapper.cs	219
Assets/Scripts/Logging and Visualization/ SequenceChartPythonProcess.cs	219
Assets/Scripts/Project Management/ ProjectManager.cs	219
Assets/Scripts/Project Management/ SaveAndLoadAOIs.cs	219
Assets/Scripts/Video and Timeline/ ClickableOnTimeline.cs	219
Assets/Scripts/Video and Timeline/ DisplayAsFrameOnTimeline.cs	220

Assets/Scripts/Video and Timeline/ DraggableOnTimeline.cs	220
Assets/Scripts/Video and Timeline/ Playhead.cs	220
Assets/Scripts/Video and Timeline/ Timeline.cs	220
Assets/Scripts/Video and Timeline/ VideoLoader.cs	220
Assets/Scripts/Video and Timeline/ VideoManager.cs	220
Assets/Scripts/VR Management/ PreVideoTestScene.cs	221
Assets/Scripts/VR Management/ VRButton.cs	221
Assets/Scripts/VR Management/ VRCamera.cs	221
Assets/Scripts/VR Management/ VRManager.cs	221
Assets/Scripts/VR Management/ VrSettingsMenu.cs	221

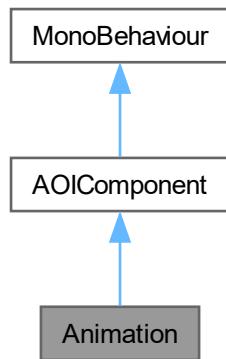
Chapter 4

Class Documentation

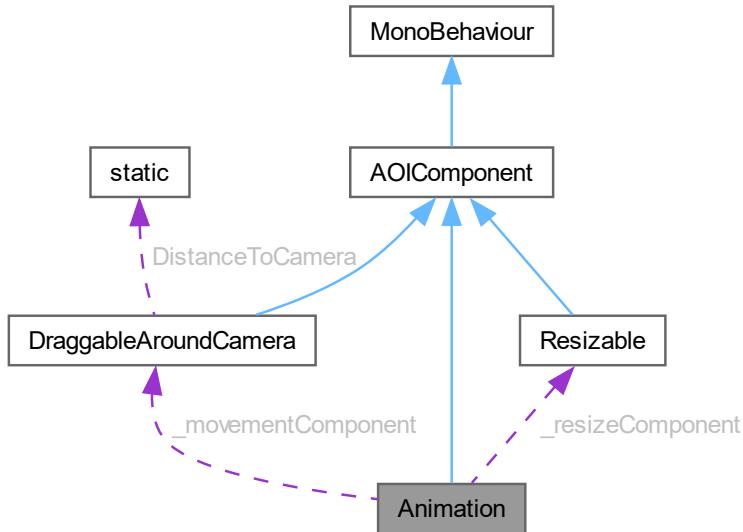
4.1 Animation Class Reference

This script does a lot of things and should probably be refactored into multiple classes. a) Creates & Updates Keyframes for position, size and shape of AOI Does this by listening to DraggableAroundCameras and Resizables events b) Calculates position, size and shape for Frames between Keyframes using interpolation c) Animates object based on current frame of video by setting position, size and shape according to calculated frames.

Inheritance diagram for Animation:



Collaboration diagram for Animation:



Classes

- class [FrameData](#)

Public Member Functions

- void [LoadKeyframes](#) (List<[FrameData](#)> loaded_keyframes)

Called by [SaveAndLoadAOIs](#) class when loading AOIs from csv. Stores passed list as keyframes, creates UI representations for all keyframes and calls [CalculateInterpolatedFrames](#).

Public Attributes

- GameObject [KeyframePrefab](#)

Protected Member Functions

- override void [Start](#) ()
- override void [OnDestroy](#) ()
- override void [OnActivate](#) ()

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.

- override void [OnDeactivate](#) ()

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.

Protected Member Functions inherited from [AOIComponent](#)

- virtual void [Start \(\)](#)
- virtual void [ActiveAOIchanged \(GameObject newAoi\)](#)
Callback for AOIMangers NewActiveAOI-Event. Called whenever user clicks on new AOI.
- abstract void [OnActivate \(\)](#)
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.
- abstract void [OnDeactivate \(\)](#)
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.
- virtual void [OnDestroy \(\)](#)

Protected Attributes

- [Resizable _resizeComponent](#)
- [DraggableAroundCamera _movementComponent](#)

Properties

- List< [FrameData](#) > [Keyframes](#) = new() [get, private set]
- List< [FrameData](#) > [InterpolatedFrames](#) = new() [get, private set]

Private Member Functions

- void [Awake \(\)](#)
- void [SetupRightClickMenu \(bool currentVisibility\)](#)
Adds visibility toggle option to right click menu of AOI.
- void [ToggleKeyframeVisibilityFromRightClickMenu \(\)](#)
Callback for AOI-Visibility Button in right-click menu of AOI.
- void [Update \(\)](#)
- bool [IsObjectInCorrectState \(\)](#)
Checks if objects state already matches correct state for current frame. Uses Timelines TargetFrame rather than VideoPlayers current frame, because current frame jumps around due to scrubbing implementation.
- void [SetObjectsStateToFrameData \(\)](#)
Sets AOIs state to state defined in Keyframe or InterpolatedFrame.
- void [CalculateInterpolatedFrames \(\)](#)
Calculates Position, Rotation and Vertex positions for all frames based on Keyframes.
- void [ObjectChangeStarted \(\)](#)
little bit unnecessary, but here just in case in the future some other things need to happen on start of object change.
- void [UpdateOrCreateKeyframeOnObjectChangeCompleted \(\)](#)
Callback for resize and movement events. Creates or updates keyframe after user has completed object change.
- void [CreateKeyframe \(\)](#)
Creates new Keyframe at current frame of timeline. Creates both UI representation as well as keyframe data.
- void [DeleteKeyframeFromRightClickMenu \(\)](#)
Callback for delete option in right click menu of keyframe right click menu.
- void [UpdateKeyframe \(int i\)](#)
Updates Keyframe at index i with objects current Position, Rotation and Vertex Information.
- void [HideKeyframes \(\)](#)
Hides objects keyframes from timeline.
- void [ShowKeyframes \(\)](#)

Displays objects keyframes on timeline.

- bool `IsFrameInKeyframes` (long frame)

Helper function checks if specified videoframe has keyframe.

- int `GetKeyframeIndexByFrame` (long frame)

Helper function that for a given videoframe returns the index of the keyframe. Returns -1 if no keyframe exists for videoframe.

- void `SetKeyframeVisibility` (long frame, bool newvisibility)

Method used to change AOI at keyframe from visible to invisible. Changes AOI Tag as well as keyframe color based on new visibility.

Private Attributes

- bool `_objectIsBeingChanged` = false

4.1.1 Detailed Description

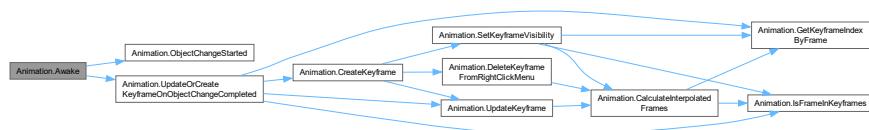
This script does a lot of things and should probably be refactored into multiple classes. a) Creates & Updates Keyframes for position, size and shape of AOI Does this by listening to DraggableAroundCameras and Resizables events b) Calculates position, size and shape for Frames between Keyframes using interpolation c) Animates object based on current frame of video by setting position, size and shape according to calculated frames.

4.1.2 Member Function Documentation

4.1.2.1 Awake()

```
void Animation.Awake ( ) [private]
```

Here is the call graph for this function:

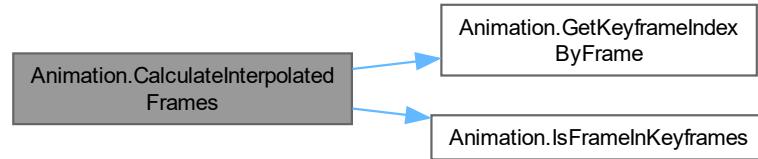


4.1.2.2 CalculateInterpolatedFrames()

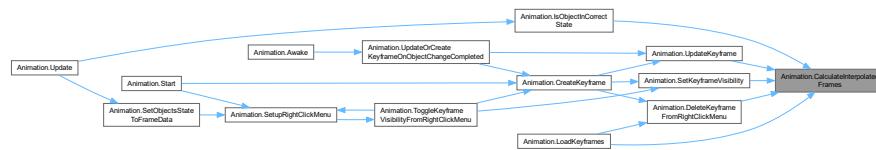
```
void Animation.CalculateInterpolatedFrames ( ) [private]
```

Calculates Position, Rotation and Vertex positions for all frames based on Keyframes.

Here is the call graph for this function:



Here is the caller graph for this function:

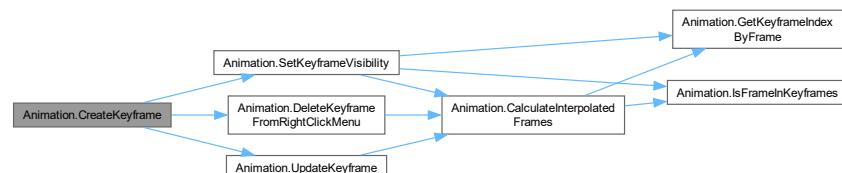


4.1.2.3 CreateKeyframe()

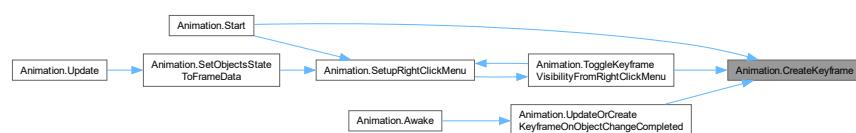
```
void Animation.CreateKeyframe( ) [private]
```

Creates new Keyframe at current frame of timeline. Creates both UI representation as well as keyframe data.

Here is the call graph for this function:



Here is the caller graph for this function:

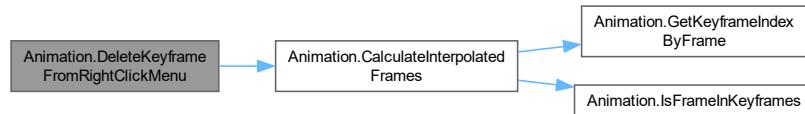


4.1.2.4 DeleteKeyframeFromRightClickMenu()

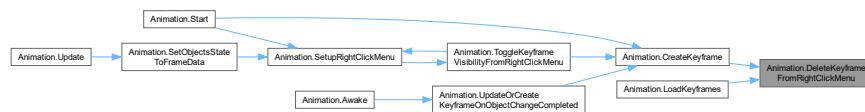
```
void Animation.DeleteKeyframeFromRightClickMenu ( ) [private]
```

Callback for delete option in right click menu of keyframe right click menu.

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.2.5 GetKeyframeIndexByFrame()

```
int Animation.GetKeyframeIndexByFrame (
    long frame ) [private]
```

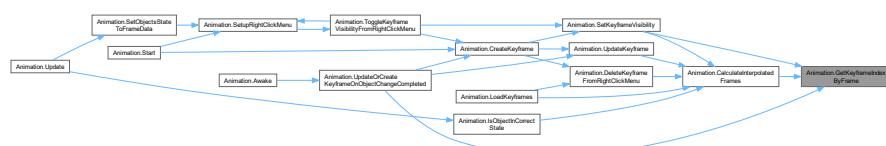
Helper function that for a given videoframe returns the index of the keyframe. Returns -1 if no keyframe exists for videoframe.

Parameters

<i>frame</i>	Videoframe for which to get index of keyframe.
--------------	--

Returns

Here is the caller graph for this function:



4.1.2.6 HideKeyframes()

```
void Animation.HideKeyframes ( ) [private]
```

Hides objects keyframes from timeline.

Here is the caller graph for this function:



4.1.2.7 IsFrameInKeyframes()

```
bool Animation.IsFrameInKeyframes (
    long frame ) [private]
```

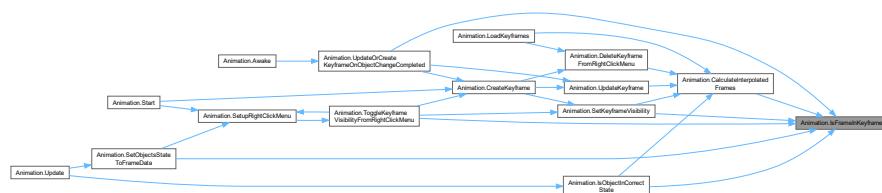
[Helper](#) function checks if specified videoframe has keyframe.

Parameters

frame	Videoframe to check for keyframe.
-------	-----------------------------------

Returns

Here is the caller graph for this function:



4.1.2.8 IsObjectInCorrectState()

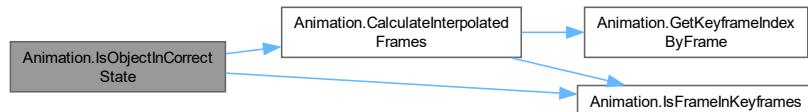
```
bool Animation.IsObjectInCorrectState ( ) [private]
```

Checks if objects state already matches correct state for current frame. Uses Timelines TargetFrame rather than VideoPlayers current frame, because current frame jumps around due to scrubbing implementation.

Returns

Returns true if object is already in correct state, false if not.

Here is the call graph for this function:



Here is the caller graph for this function:

**4.1.2.9 LoadKeyframes()**

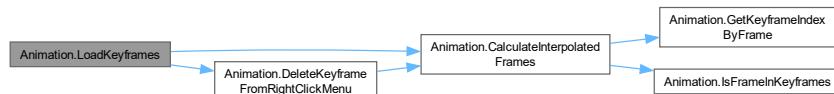
```
void Animation.LoadKeyframes (
    List< FrameData > loaded_keyframes )
```

Called by [SaveAndLoadAOIs](#) class when loading AOIs from csv. Stores passed list as keyframes, creates UI representations for all keyframes and calls CalculateInterpolatedFrames.

Parameters

<i>loaded_keyframes</i>	List of loaded Keyframes that were stored in csv.
-------------------------	---

Here is the call graph for this function:

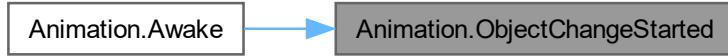


4.1.2.10 ObjectChangeStarted()

```
void Animation.ObjectChangeStarted ( ) [private]
```

little bit unnecessary, but here just in case in the future some other things need to happen on start of object change.

Here is the caller graph for this function:



4.1.2.11 OnActivate()

```
override void Animation.OnActivate ( ) [protected], [virtual]
```

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.

Implements [AOIComponent](#).

Here is the call graph for this function:



4.1.2.12 OnDeactivate()

```
override void Animation.OnDeactivate ( ) [protected], [virtual]
```

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.

Implements [AOIComponent](#).

Here is the call graph for this function:



4.1.2.13 OnDestroy()

```
override void Animation.OnDestroy ( ) [protected], [virtual]
```

Reimplemented from [AOIComponent](#).

4.1.2.14 SetKeyframeVisibility()

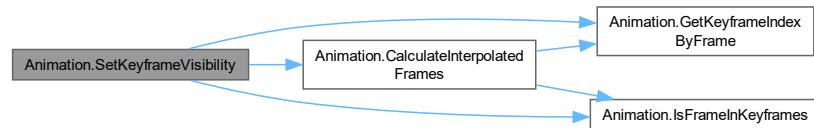
```
void Animation.SetKeyframeVisibility (
    long frame,
    bool newvisibility ) [private]
```

Method used to change AOI at keyframe from visible to invisible. Changes AOI Tag as well as keyframe color based on new visibility.

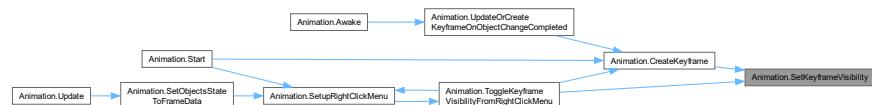
Parameters

<i>frame</i>	Position of keyframe to change visibility of.
<i>newvisibility</i>	Visibility to change keyframe to.

Here is the call graph for this function:



Here is the caller graph for this function:

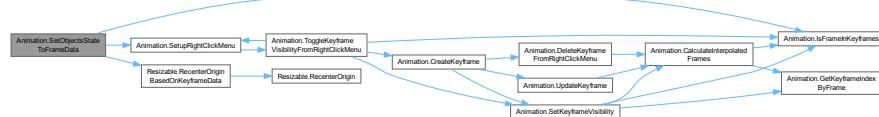


4.1.2.15 SetObjectsStateToFrameData()

```
void Animation.SetObjectsStateToFrameData ( ) [private]
```

Sets AOIs state to state defined in Keyframe or InterpolatedFrame.

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.2.16 SetupRightClickMenu()

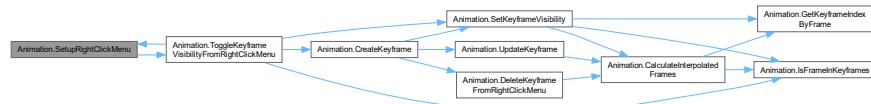
```
void Animation.SetupRightClickMenu (
    bool currentVisibility ) [private]
```

Adds visibility toggle option to right click menu of AOI.

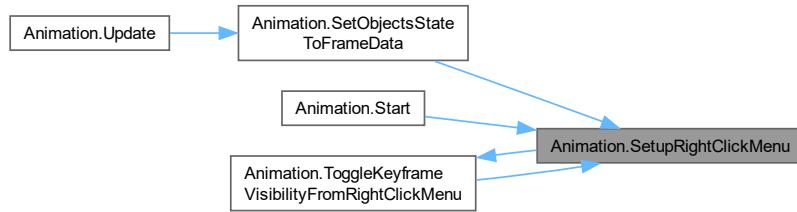
Parameters

<i>currentVisibility</i>	Information whether or not AOI is currently visible to eyetracking raycast
--------------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.2.17 ShowKeyframes()

```
void Animation.ShowKeyframes( ) [private]
```

Displays objects keyframes on timeline.

Here is the caller graph for this function:

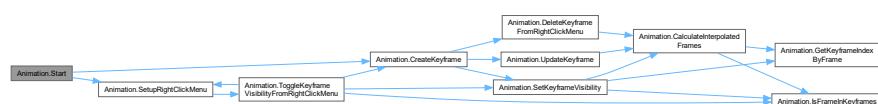


4.1.2.18 Start()

```
override void Animation.Start( ) [protected], [virtual]
```

Reimplemented from [AOIComponent](#).

Here is the call graph for this function:

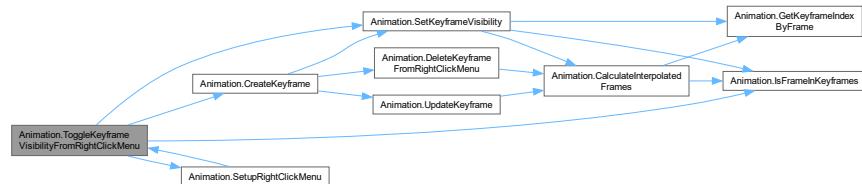


4.1.2.19 ToggleKeyframeVisibilityFromRightClickMenu()

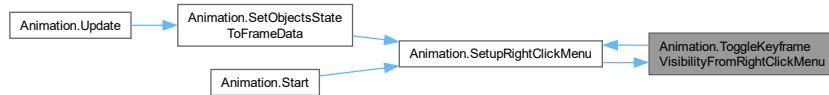
```
void Animation.ToggleKeyframeVisibilityFromRightClickMenu ( ) [private]
```

Callback for AOI-Visibility Button in right-click menu of AOI.

Here is the call graph for this function:

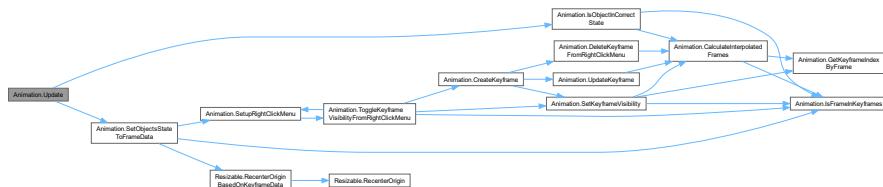


Here is the caller graph for this function:

**4.1.2.20 Update()**

```
void Animation.Update ( ) [private]
```

Here is the call graph for this function:

**4.1.2.21 UpdateKeyframe()**

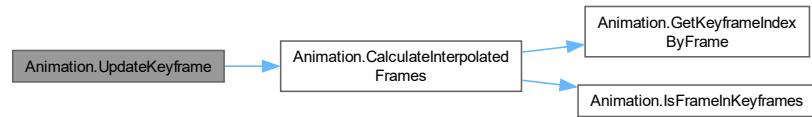
```
void Animation.UpdateKeyframe (
    int i ) [private]
```

Updates Keyframe at index i with objects current Position, Rotation and Vertex Information.

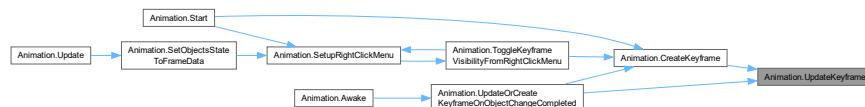
Parameters

<i>i</i>	Index of keyframe in list to be updated.
----------	--

Here is the call graph for this function:



Here is the caller graph for this function:

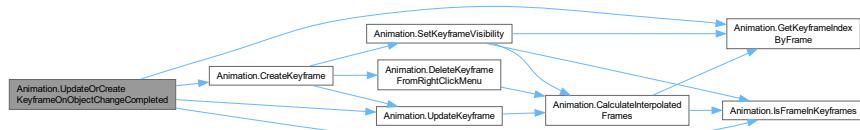


4.1.2.22 UpdateOrCreateKeyframeOnObjectChangeCompleted()

```
void Animation.UpdateOrCreateKeyframeOnObjectChangeCompleted ( ) [private]
```

Callback for resize and movement events. Creates or updates keyframe after user has completed object change.

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.3 Member Data Documentation

4.1.3.1 `_movementComponent`

```
DraggableAroundCamera Animation._movementComponent [protected]
```

4.1.3.2 `_objectIsBeingChanged`

```
bool Animation._objectIsBeingChanged = false [private]
```

4.1.3.3 `_resizeComponent`

```
Resizable Animation._resizeComponent [protected]
```

4.1.3.4 `KeyframePrefab`

```
GameObject Animation.KeyframePrefab
```

4.1.4 Property Documentation

4.1.4.1 `InterpolatedFrames`

```
List<FrameData> Animation.InterpolatedFrames = new() [get], [private set]
```

4.1.4.2 `Keyframes`

```
List<FrameData> Animation.Keyframes = new() [get], [private set]
```

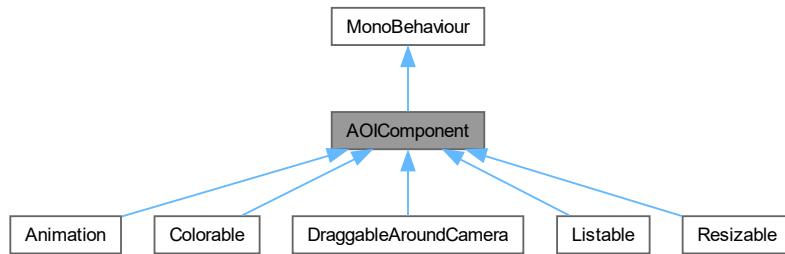
The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/AOI Components/[Animation.cs](#)

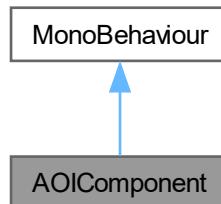
4.2 AOIComponent Class Reference

This is the base class for components that make up an area of interest. The user can highlight and work with one AOI at a time, marking that one as active (done by the AOI Manager) This class subscribes to the event of changing the active AOI and specifies the relevant methods (OnActivate, OnDeactivate) that need to be implemented by derived classes.

Inheritance diagram for AOIComponent:



Collaboration diagram for AOIComponent:



Protected Member Functions

- virtual void [Start \(\)](#)
- virtual void [ActiveAOIchanged](#) (GameObject newAoi)

Callback for AOIManagers NewActiveAOI-Event. Called whenever user clicks on new AOI.
- abstract void [OnActivate \(\)](#)

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.
- abstract void [OnDeactivate \(\)](#)

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.
- virtual void [OnDestroy \(\)](#)

4.2.1 Detailed Description

This is the base class for components that make up an area of interest. The user can highlight and work with one AOI at a time, marking that one as active (done by the AOI Manager) This class subscribes to the event of changing the active AOI and specifies the relevant methods (OnActivate, OnDeactivate) that need to be implemented by derived classes.

4.2.2 Member Function Documentation

4.2.2.1 ActiveAOIchanged()

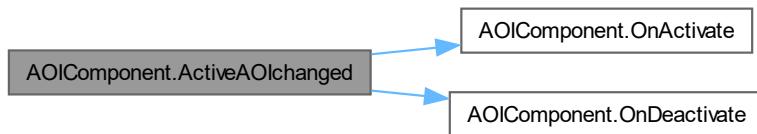
```
virtual void AOIComponent.ActiveAOIchanged (
    GameObject newAoi ) [protected], [virtual]
```

Callback for AOIMangers NewActiveAOI-Event. Called whenever user clicks on new AOI.

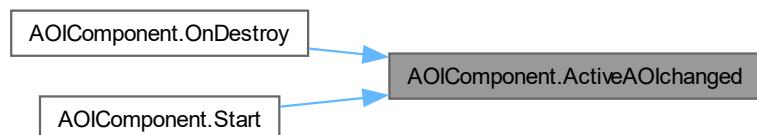
Parameters

<i>newAoi</i>	AOI that user has clicked on. New active AOI.
---------------	---

Here is the call graph for this function:



Here is the caller graph for this function:



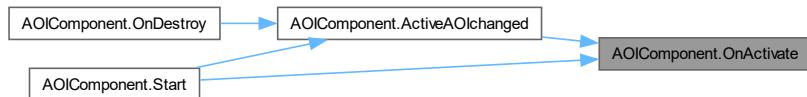
4.2.2.2 OnActivate()

```
abstract void AOIComponent.OnActivate ( ) [protected], [pure virtual]
```

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.

Implemented in [Animation](#), [Colorable](#), [DraggableAroundCamera](#), [Listable](#), and [Resizable](#).

Here is the caller graph for this function:



4.2.2.3 OnDeactivate()

```
abstract void AOIComponent.OnDeactivate ( ) [protected], [pure virtual]
```

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.

Implemented in [Animation](#), [Colorable](#), [DraggableAroundCamera](#), [Listable](#), and [Resizable](#).

Here is the caller graph for this function:



4.2.2.4 OnDestroy()

```
virtual void AOIComponent.OnDestroy ( ) [protected], [virtual]
```

Reimplemented in [Animation](#), and [Listable](#).

Here is the call graph for this function:

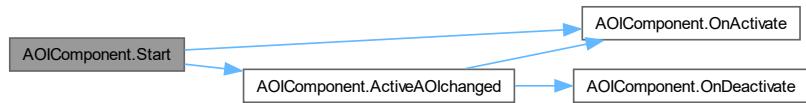


4.2.2.5 Start()

```
virtual void AOIComponent.Start ( ) [protected], [virtual]
```

Reimplemented in [Animation](#), [Colorable](#), [Listable](#), and [Resizable](#).

Here is the call graph for this function:



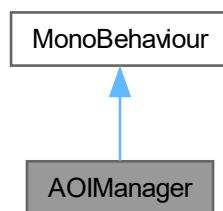
The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/AOI Components/[AOIComponent.cs](#)

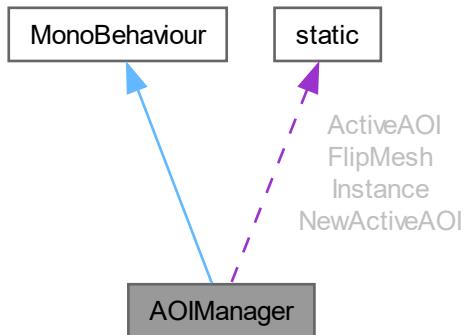
4.3 AOIManager Class Reference

This class does 3 things 1.Manages AOIs based on user input a) Deactivates currently active AOIs and spawns new AOIs if user clicks on empty space in 360 degree video b) Sets an AOI as active when user clicks on AOI in 360 degree video and deactivates the previously selected AOI.

Inheritance diagram for AOIManager:



Collaboration diagram for AOIManager:



Public Member Functions

- void [ToggleAOIColliders](#) (bool target_collider_state)
Searches for all AOI Objects and disabled or enables their colliders based on target_collider_state parameter.
- GameObject [SpawnAoi](#) (Vector3 spawnPosition)
instantiates new AOIPrefab object at specified spawn position.

Static Public Member Functions

- static void [SetActiveObject](#) (GameObject newactive)
Sets active AOI after use has clicked a new AOI and invokes NewActiveAOI Event.

Public Attributes

- GameObject [AOIPrefab](#)
- float [SpawnDistance](#) = 10f

Static Public Attributes

- static [AOIManager](#) [Instance](#)
- static GameObject [ActiveAOI](#) = null
- static UnityEvent< GameObject > [NewActiveAOI](#) = new()
- static bool [FlipMesh](#) = true

Private Member Functions

- void [Awake \(\)](#)
- void [Start \(\)](#)
- void [OnVrStatusChange \(VRManager.VRStatus status\)](#)

callback for VRManagers status change event. When VR is being enabled or disabled it toggles AOI Colliders based on whether the TestScene is being used.
- void [OnTestSceneToggled \(bool isEnabled\)](#)

Callback for VrSettingsMenu TestScene toggle. Reacts to user toggling whether or not they want to use a testscene before displaying the 360 degree video.
- void [Update \(\)](#)

The Update function is currently pretty complicated, because it not only handles the spawning of new AOIs, but also handles what AOI/AOI-Handle receives MouseDown events when multiple AOIs overlap.
- void [StartManualResizingDueToOverlap \(GameObject resizingObject\)](#)

If elements overlap this method sends the OnMouseDown event to the correct scripts manually.
- void [CompleteManualResizingDueToOverlap \(\)](#)

After resizing is completed this method manually calls the OnMouseUp-Event on all relevant components of the last object that was resized.
- void [SpawnAOlatCursorPosition \(\)](#)

Calculates worldspace-position from cursor for spawning a new AOI.

Private Attributes

- GameObject [_canvas](#)
- bool [_manuallyTriggeredResizeDueToOverlap = false](#)
- GameObject [_objectToManuallyResize = null](#)
- bool [_manuallyTriggeredMovementDueToOverlap = false](#)

4.3.1 Detailed Description

This class does 3 things 1.Manages AOIs based on user input a) Deactivates currently active AOIs and spawns new AOIs if user clicks on empty space in 360 degree video b) Sets an AOI as active when user clicks on AOI in 360 degree video and deactivates the previously selected AOI.

1. Manages AOI component behaviour (specifically resizing and movement of AOIs) when multiple AOIs overlap
 - a) prioritizes active AOI over inactive AOIs b) prioritizes resizing over movement
2. Disables AOIs during PreVideoStimulus TestScene so that editor doesnt accidentally edit AOIs while they're not visible.

4.3.2 Member Function Documentation

4.3.2.1 Awake()

```
void AOIManager.Awake ( ) [private]
```

4.3.2.2 CompleteManualResizingDueToOverlap()

```
void AOIManager.CompleteManualResizingDueToOverlap ( ) [private]
```

After resizing is completed this method manually calls the OnMouseUp-Event on all relevant components of the last object that was resized.

Here is the caller graph for this function:



4.3.2.3 OnTestSceneToggled()

```
void AOIManager.OnTestSceneToggled (
    bool isEnabled) [private]
```

Callback for [VrSettingsMenu](#) TestScene toggle. Reacts to user toggling whether or not they want to use a testscene before displaying the 360 degree video.

Parameters

isEnabled

Here is the call graph for this function:



Here is the caller graph for this function:



4.3.2.4 OnVrStatusChange()

```
void AOIManager.OnVrStatusChange (
    VRManager::VRStatus status ) [private]
```

callback for VRManagers status change event. When VR is being enabled or disabled it toggles AOI Colliders based on whether the TestScene is being used.

Parameters

<i>status</i>	New VR Status
---------------	---------------

Here is the call graph for this function:



Here is the caller graph for this function:



4.3.2.5 SetActiveObject()

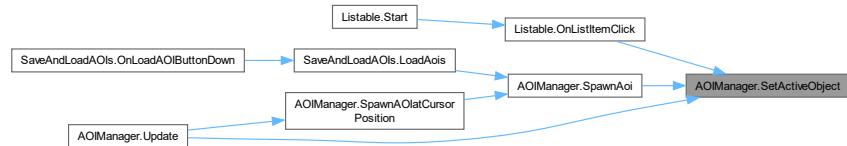
```
static void AOIManager.SetActiveObject (
    GameObject newactive ) [static]
```

Sets active AOI after user has clicked a new AOI and invokes NewActiveAOI Event.

Parameters

<i>newactive</i>	The last Object the user has clicked that should be set as active.
------------------	--

Here is the caller graph for this function:



4.3.2.6 `SpawnAoi()`

```
GameObject AOIManager.SpawnAoi (
    Vector3 spawnPosition )
```

instantiates new AOIPrefab object at specified spawn position.

Parameters

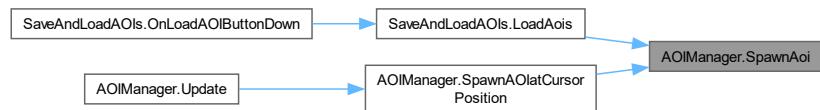
<code>spawnPosition</code>	Position to instantiate new object at.
----------------------------	--

Returns

Here is the call graph for this function:



Here is the caller graph for this function:



4.3.2.7 SpawnAOIatCursorPosition()

```
void AOIManager.SpawnAOIatCursorPosition ( ) [private]
```

Calculates worldspace-position from cursor for spawning a new AOI.

Here is the call graph for this function:



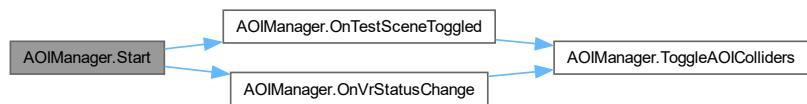
Here is the caller graph for this function:



4.3.2.8 Start()

```
void AOIManager.Start ( ) [private]
```

Here is the call graph for this function:



4.3.2.9 StartManualResizingDueToOverlap()

```
void AOIManager.StartManualResizingDueToOverlap (
    GameObject resizingObject ) [private]
```

If elements overlap this method sends the OnMouseDown event to the correct scripts manually.

Parameters

<i>resizingObject</i>	Object which components manually need to receive OnMouseDown-Events
-----------------------	---

Here is the caller graph for this function:



4.3.2.10 ToggleAOIColliders()

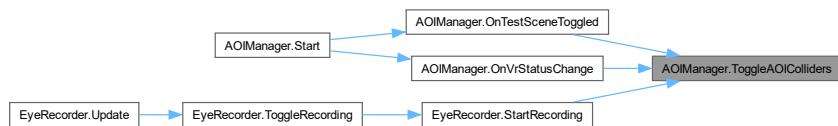
```
void AOIManager.ToggleAOIColliders (
    bool target.collider.state )
```

Searches for all AOI Objects and disabled or enables their colliders based on target.collider.state parameter.

Parameters

<i>target.collider.state</i>	New target state for colliders, true = enabled, false = disabled.
------------------------------	---

Here is the caller graph for this function:

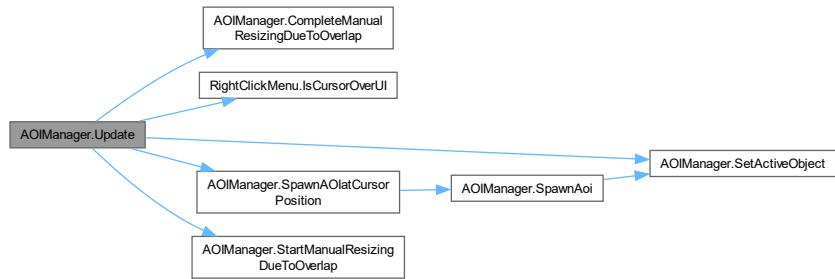


4.3.2.11 Update()

```
void AOIManager.Update ( ) [private]
```

The Update function is currently pretty complicated, because it not only handles the spawning of new AOIs, but also handles what AOI/AOI-Handle receives MouseDown events when multiple AOIs overlap.

Here is the call graph for this function:



4.3.3 Member Data Documentation

4.3.3.1 `_canvas`

```
GameObject AOIManager._canvas [private]
```

4.3.3.2 `_manuallyTriggeredMovementDueToOverlap`

```
bool AOIManager._manuallyTriggeredMovementDueToOverlap = false [private]
```

4.3.3.3 `_manuallyTriggeredResizeDueToOverlap`

```
bool AOIManager._manuallyTriggeredResizeDueToOverlap = false [private]
```

4.3.3.4 `_objectToManuallyResize`

```
GameObject AOIManager._objectToManuallyResize = null [private]
```

4.3.3.5 `ActiveAOI`

```
GameObject AOIManager.ActiveAOI = null [static]
```

4.3.3.6 `AOIPrefab`

```
GameObject AOIManager.AOIPrefab
```

4.3.3.7 `FlipMesh`

```
bool AOIManager.FlipMesh = true [static]
```

4.3.3.8 Instance

```
AOIManager AOIManager.Instance [static]
```

4.3.3.9 NewActiveAOI

```
UnityEvent<GameObject> AOIManager.NewActiveAOI = new() [static]
```

4.3.3.10 SpawnDistance

```
float AOIManager.SpawnDistance = 10f
```

The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/AOIManager.cs

4.4 EventDetection.AoiParameters Class Reference

Public Member Functions

- [AoiParameters \(string aoiName\)](#)

Public Attributes

- string [AoiName](#)
- int [DwellCount](#) = 0
- int [EntryTime](#)
- int [FirstPassDwellTime](#)
- int [TotalDwellTime](#) = 0
- float [ProportionOfTotalDwellTime](#)
- int [FixationCount](#) = 0
- int [TimeToFirstFixation](#)
- int [DurationOfFirstFixation](#)
- float [ProportionOfFixations](#)

4.4.1 Constructor & Destructor Documentation

4.4.1.1 AoiParameters()

```
EventDetection.AoiParameters.AoiParameters (
    string aoiName )
```

4.4.2 Member Data Documentation

4.4.2.1 AoiName

```
string EventDetection.AoiParameters.AoiName
```

4.4.2.2 DurationOfFirstFixation

```
int EventDetection.AoiParameters.DurationOfFirstFixation
```

4.4.2.3 DwellCount

```
int EventDetection.AoiParameters.DwellCount = 0
```

4.4.2.4 EntryTime

```
int EventDetection.AoiParameters.EntryTime
```

4.4.2.5 FirstPassDwellTime

```
int EventDetection.AoiParameters.FirstPassDwellTime
```

4.4.2.6 FixationCount

```
int EventDetection.AoiParameters.FixationCount = 0
```

4.4.2.7 ProportionOfFixations

```
float EventDetection.AoiParameters.ProportionOfFixations
```

4.4.2.8 ProportionOfTotalDwellTime

```
float EventDetection.AoiParameters.ProportionOfTotalDwellTime
```

4.4.2.9 TimeToFirstFixation

```
int EventDetection.AoiParameters.TimeToFirstFixation
```

4.4.2.10 TotalDwellTime

```
int EventDetection.AoiParameters.TotalDwellTime =0
```

The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/[EventDetection.cs](#)

4.5 SaveAndLoadAOIs.AOISaveData Class Reference

Public Attributes

- string **Name**
- List<[Animation.FrameData](#) > **Keyframes**

4.5.1 Member Data Documentation

4.5.1.1 Keyframes

```
List<Animation.FrameData> SaveAndLoadAOIs.AOISaveData.Keyframes
```

4.5.1.2 Name

```
string SaveAndLoadAOIs.AOISaveData.Name
```

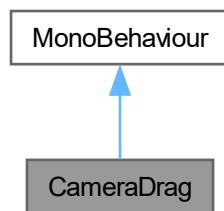
The documentation for this class was generated from the following file:

- Assets/Scripts/Project Management/[SaveAndLoadAOIs.cs](#)

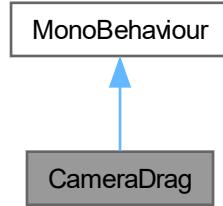
4.6 CameraDrag Class Reference

This script is attached to the Editor Camera and allows the User to change the portion of the 360 degree video being displayed, by holding down right click and dragging. The camera then rotates based on the cursor movement.

Inheritance diagram for CameraDrag:



Collaboration diagram for CameraDrag:



Public Attributes

- float `Speed` = 3.5f

Private Member Functions

- void `Start()`
- void `Update()`

Since this is a very simple script all functionality is implemented in update function.

Private Attributes

- `GameObject _canvas`
- float `_X`
- float `_Y`

4.6.1 Detailed Description

This script is attached to the Editor Camera and allows the User to change the portion of the 360 degree video being displayed, by holding down right click and dragging. The camera then rotates based on the cursor movement.

4.6.2 Member Function Documentation

4.6.2.1 Start()

```
void CameraDrag.Start( ) [private]
```

4.6.2.2 Update()

```
void CameraDrag.Update ( ) [private]
```

Since this is a very simple script all functionality is implemented in update function.

Here is the call graph for this function:



4.6.3 Member Data Documentation

4.6.3.1 _canvas

```
GameObject CameraDrag._canvas [private]
```

4.6.3.2 _X

```
float CameraDrag._X [private]
```

4.6.3.3 _Y

```
float CameraDrag._Y [private]
```

4.6.3.4 Speed

```
float CameraDrag.Speed = 3.5f
```

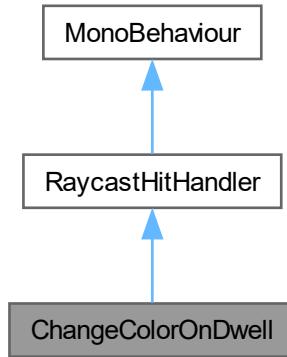
The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/[CameraDrag.cs](#)

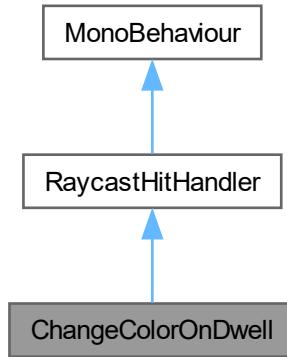
4.7 ChangeColorOnDwell Class Reference

This script inherits from the [RaycastHitHandler](#) and therefore its functions are called by the [GazeRaycaster](#). It simply changes the color of the object it is attached to when the gaze of the person wearing the vr headset is over the object.

Inheritance diagram for ChangeColorOnDwell:



Collaboration diagram for ChangeColorOnDwell:



Public Member Functions

- `override void OnRaycastEnter ()`
Change color on gaze-raycast enter.
- `override void OnRaycastExit ()`

Change color on gaze-raycast exit.

- virtual void [OnRaycastEnter \(\)](#)
Called by GazeRaycaster when Dwell on object is started.
- virtual void [OnRaycastExit \(\)](#)
Called by GazeRaycaster when Dwell on object has ended.

Public Attributes

- Color [RaycastEnterColor](#) = Color.blue
- Color [RaycastExitColor](#) = Color.white

Private Member Functions

- void [Start \(\)](#)

Private Attributes

- Renderer [_renderer](#)

Additional Inherited Members

Properties inherited from [RaycastHitHandler](#)

- bool [IsCurrentlyHitByRaycast](#) = false [get, private set]

4.7.1 Detailed Description

This script inherits from the [RaycastHitHandler](#) and therefore its functions are called by the [GazeRaycaster](#). It simply changes the color of the object it is attached to when the gaze of the person wearing the vr headset is over the object.

4.7.2 Member Function Documentation

4.7.2.1 [OnRaycastEnter\(\)](#)

```
override void ChangeColorOnDwell.OnRaycastEnter ( ) [virtual]
```

Change color on gaze-raycast enter.

Reimplemented from [RaycastHitHandler](#).

4.7.2.2 OnRaycastExit()

```
override void ChangeColorOnDwell.OnRaycastExit ( ) [virtual]
```

Change color on gaze-raycast exit.

Reimplemented from [RaycastHitHandler](#).

4.7.2.3 Start()

```
void ChangeColorOnDwell.Start ( ) [private]
```

4.7.3 Member Data Documentation

4.7.3.1 _renderer

```
Renderer ChangeColorOnDwell._renderer [private]
```

4.7.3.2 RaycastEnterColor

```
Color ChangeColorOnDwell.RaycastEnterColor = Color.blue
```

4.7.3.3 RaycastExitColor

```
Color ChangeColorOnDwell.RaycastExitColor = Color.white
```

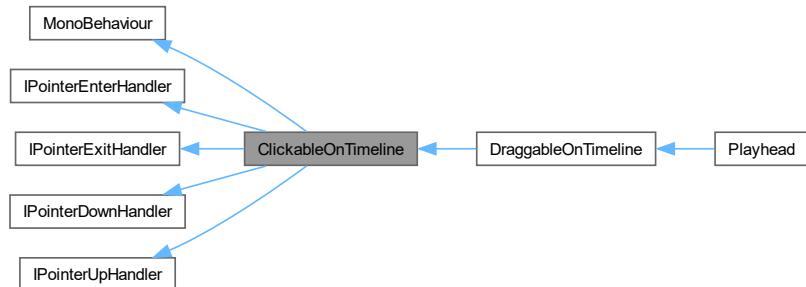
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/[ChangeColorOnDwell.cs](#)

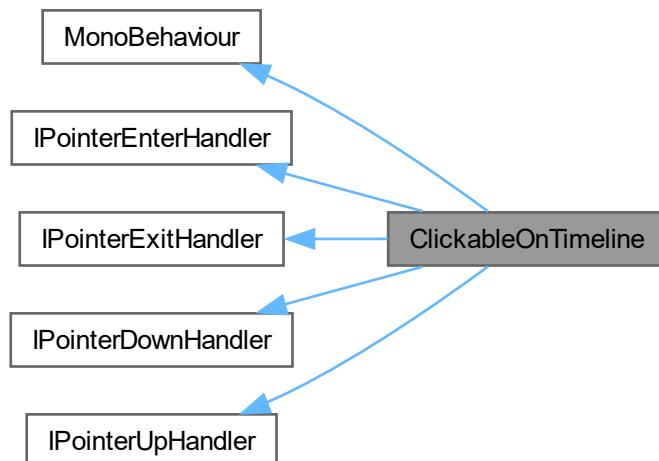
4.8 ClickableOnTimeline Class Reference

This class serves as a base for all clickable children of the timeline. it makes sure that the hover state between the timeline and its children is properly handed-over. For example if the user clicks on a keyframe and then drags the cursor, that should not move the playhead on the timeline because the user wanted to interact with the keyframe, not the timeline. Vice versa if the user scrubs the playhead across the timeline hovering over a keyframe should not trigger a click on the keyframe.

Inheritance diagram for ClickableOnTimeline:



Collaboration diagram for ClickableOnTimeline:



Public Member Functions

- virtual void [OnPointerEnter](#) (PointerEventData eventData)

When the cursor hover above an item on the timeline the item marks itself as hovered.
- virtual void [OnPointerExit](#) (PointerEventData eventData)

If the cursor leaves the area of the clickable item but has previously been clicked and the mouse is still pressed down, the item makes sure that the timeline still doesn't register as being hovered, because the user is still interacting with the item itself, since the mouse is still pressed down (the user might e.g. be trying to drag an item across the timeline.). If the cursor leaves the area of the item and the item has not been clicked, it sets itself as not hovered and hands over the hover to the timeline.
- virtual void [OnPointerDown](#) (PointerEventData eventData)

If an item on the timeline has been clicked it marks itself as clicked and takes the hover away from the timeline.
- virtual void [OnPointerUp](#) (PointerEventData eventData)

On Pointer up the item registers itself as not clicked anymore, allows edge snapping of the timeline and hands over the hover to the timeline.

Protected Member Functions

- virtual void [Start \(\)](#)
- virtual void [Update \(\)](#)

After a clickable item on the timeline has been clicked, for as long as the mouse is still down, the update function makes sure that the timeline never registers as being "hovered", since the user is hovering and item on the timeline, not the timeline itself.

Protected Attributes

- RectTransform [_rect](#)

Properties

- [Timeline Timeline \[get, protected set\]](#)
- bool [IsHovered = false \[get, protected set\]](#)
- bool [IsClickedAndMouseStillDown = false \[get, protected set\]](#)

4.8.1 Detailed Description

This class serves as a base for all clickable children of the timeline. it makes sure that the hover state between the timeline and its children is properly handed-over. For example if the user clicks on a keyframe and then drags the cursor, that should not move the playhead on the timeline because the user wanted to interact with the keyframe, not the timeline. Vice versa if the user scrubs the playhead across the timeline hovering over a keyframe should not trigger a click on the keyframe.

4.8.2 Member Function Documentation

4.8.2.1 OnPointerDown()

```
virtual void ClickableOnTimeline.OnPointerDown (
    PointerEventData eventData ) [virtual]
```

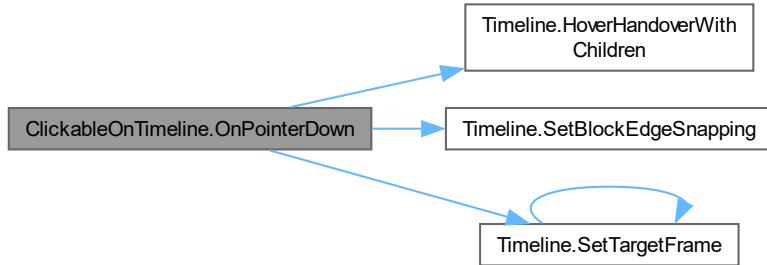
If an item on the timeline has been clicked it marks itself as clicked and takes the hover away from the timeline.

Parameters

<code>eventData</code>	<input type="text"/>
------------------------	----------------------

Reimplemented in [DraggableOnTimeline](#).

Here is the call graph for this function:



4.8.2.2 OnPointerEnter()

```
virtual void ClickableOnTimeline.OnPointerEnter (
    PointerEventData eventData ) [virtual]
```

When the cursor hover above an item on the timeline the item marks itself as hovered.

Parameters

<code>eventData</code>	<input type="button" value=""/>
------------------------	---------------------------------

4.8.2.3 OnPointerExit()

```
virtual void ClickableOnTimeline.OnPointerExit (
    PointerEventData eventData ) [virtual]
```

If the cursor leaves the area of the clickable item but has previously beed clicked and the mouse is still pressed down, the item makes sure that the timeline still doesn't register as being hovered, because the user is still interacting with the item itself, since the mouse is still pressed down (the user might e.g. be trying to drag an item across the timeline.). If the cursor leaves the area of the item and the item has not been clicked, it sets itself as not hovered and hands over the hover to the timeline.

Parameters

<code>eventData</code>	<input type="button" value=""/>
------------------------	---------------------------------

Here is the call graph for this function:



4.8.2.4 OnPointerUp()

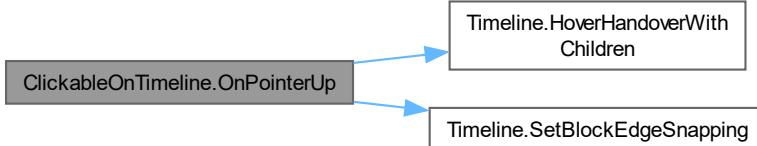
```
virtual void ClickableOnTimeline.OnPointerUp (
    PointerEventData eventData ) [virtual]
```

On Pointer up the item registers itself as not clicked anymore, allows edge snapping of the timeline abd handsover hover to the timeline.

Parameters

<i>eventData</i>	
------------------	--

Here is the call graph for this function:



4.8.2.5 Start()

```
virtual void ClickableOnTimeline.Start ( ) [protected], [virtual]
```

Reimplemented in [DraggableOnTimeline](#).

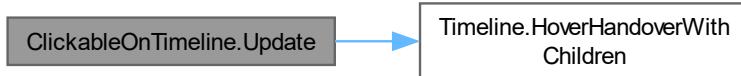
4.8.2.6 Update()

```
virtual void ClickableOnTimeline.Update ( ) [protected], [virtual]
```

After a clickable item on the timeline has been clicked, for as long as the mouse is still down, the update function makes sure that the timeline never registeres as being "hovered", since the user is hovering and item on the timeline, not the timeline itself.

Reimplemented in [DraggableOnTimeline](#), and [Playhead](#).

Here is the call graph for this function:



4.8.3 Member Data Documentation

4.8.3.1 `_rect`

```
RectTransform ClickableOnTimeline._rect [protected]
```

4.8.4 Property Documentation

4.8.4.1 `IsClickedAndMouseStillDown`

```
bool ClickableOnTimeline.IsClickedAndMouseStillDown = false [get], [protected set]
```

4.8.4.2 `IsHovered`

```
bool ClickableOnTimeline.IsHovered = false [get], [protected set]
```

4.8.4.3 `Timeline`

```
Timeline ClickableOnTimeline.Timeline [get], [protected set]
```

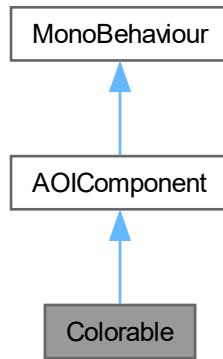
The documentation for this class was generated from the following file:

- Assets/Scripts/Video and Timeline/[ClickableOnTimeline.cs](#)

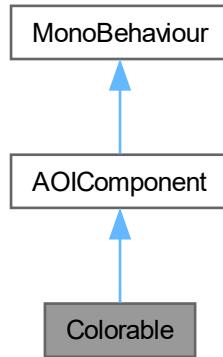
4.9 Colorable Class Reference

Class that was supposed to allow AOIs to be recolored from AOI List or [RightClickMenu](#). Not implemented though, so class is mostly useless. Currently only handles color of AOI based on whether or not AOI is selected by user.

Inheritance diagram for Colorable:



Collaboration diagram for Colorable:



Public Types

- enum [AoiColors](#) { [ACTIVE](#) , [INACTIVE](#) , [INVISBLE](#) }

Public Member Functions

- void [SetColor](#) ([AoiColors](#) color)
set color of AOI based to one of available AoiColors.

Protected Member Functions

- override void [Start \(\)](#)
- override void [OnActivate \(\)](#)
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.
- override void [OnDeactivate \(\)](#)
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.

Protected Member Functions inherited from [AOIComponent](#)

- virtual void [Start \(\)](#)
- virtual void [ActiveAOIchanged \(GameObject newAoi\)](#)
Callback for AOIMangers NewActiveAOI-Event. Called whenever user clicks on new AOI.
- abstract void [OnActivate \(\)](#)
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.
- abstract void [OnDeactivate \(\)](#)
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.
- virtual void [OnDestroy \(\)](#)

Private Member Functions

- void [SetColor \(float r, float g, float b, float a\)](#)
set color of AOI to color specified by parameters.

Private Attributes

- Renderer [_renderer](#)
- [AoiColors _color](#)

4.9.1 Detailed Description

Class that was supposed to allow AOIs to be recolored from AOI List or [RightClickMenu](#). Not implemented though, so class is mostly useless. Currently only handles color of AOI based on whether or not AOI is selected by user.

4.9.2 Member Enumeration Documentation

4.9.2.1 [AoiColors](#)

```
enum Colorable.AoiColors
```

Enumerator

ACTIVE	
INACTIVE	
INVISBLE	

4.9.3 Member Function Documentation

4.9.3.1 OnActivate()

```
override void Colorable.OnActivate ( ) [protected], [virtual]
```

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.

Implements [AOIComponent](#).

Here is the call graph for this function:



4.9.3.2 OnDeactivate()

```
override void Colorable.OnDeactivate ( ) [protected], [virtual]
```

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.

Implements [AOIComponent](#).

Here is the call graph for this function:



4.9.3.3 SetColor() [1/2]

```
void ColorableSetColor (   
    AoiColors color )
```

set color of AOI based to one of available AoiColors.

Parameters

color	
-------	--

Here is the call graph for this function:

**4.9.3.4 SetColor() [2/2]**

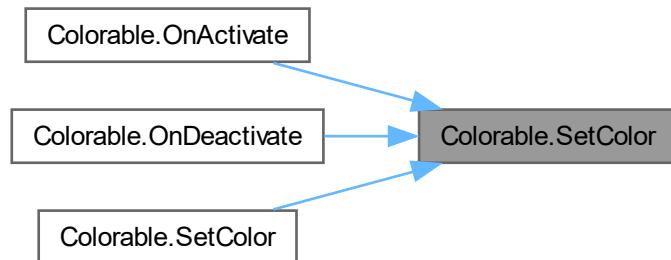
```
void ColorableSetColor (
    float r,
    float g,
    float b,
    float a )  [private]
```

set color of AOI to color specified by parameters.

Parameters

r	red
g	green
b	blue
a	alpha

Here is the caller graph for this function:



4.9.3.5 Start()

```
override void Colorable.Start ( )  [protected], [virtual]
```

Reimplemented from [AOIComponent](#).

4.9.4 Member Data Documentation

4.9.4.1 _color

```
AoiColors Colorable._color  [private]
```

4.9.4.2 _renderer

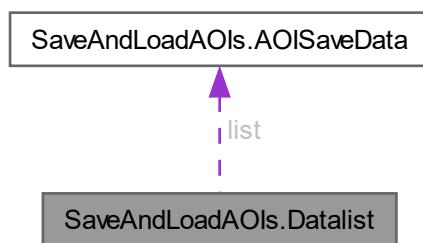
```
Renderer Colorable._renderer  [private]
```

The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/AOI Components/[Colorable.cs](#)

4.10 SaveAndLoadAOIs.Datalist Class Reference

Collaboration diagram for SaveAndLoadAOIs.Datalist:



Public Attributes

- [AOISaveData\[\] list](#)

4.10.1 Member Data Documentation

4.10.1.1 list

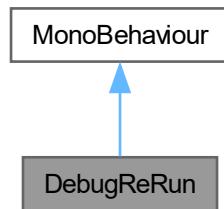
```
AOISaveData [ ] SaveAndLoadAOIs.Datalist.list
```

The documentation for this class was generated from the following file:

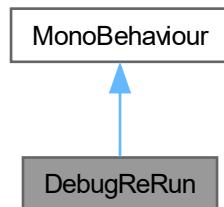
- Assets/Scripts/Project Management/[SaveAndLoadAOIs.cs](#)

4.11 DebugReRun Class Reference

Inheritance diagram for DebugReRun:



Collaboration diagram for DebugReRun:



Public Member Functions

- void [OnReRun \(\)](#)

Private Member Functions

- List< [EyeTrackingDataSample](#) > [CSVtoDataSampleList](#) (string filepath)
- void [DebugFixationSampleFlagsToCsv](#) (bool[] Fixationsample_flags, string path)

4.11.1 Member Function Documentation

4.11.1.1 CSVtoDataSampleList()

```
List< EyeTrackingDataSample > DebugReRun.CSVtoDataSampleList (
    string filepath ) [private]
```

Here is the caller graph for this function:



4.11.1.2 DebugFixationSampleFlagsToCsv()

```
void DebugReRun.DebugFixationSampleFlagsToCsv (
    bool[] Fixationsample_flags,
    string path ) [private]
```

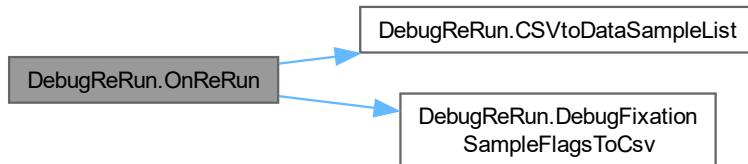
Here is the caller graph for this function:



4.11.1.3 OnReRun()

```
void DebugReRun.OnReRun( )
```

Here is the call graph for this function:



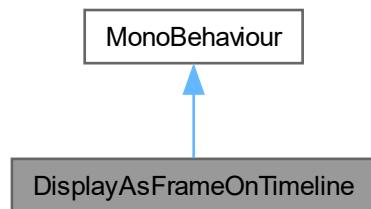
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/[DebugReRun.cs](#)

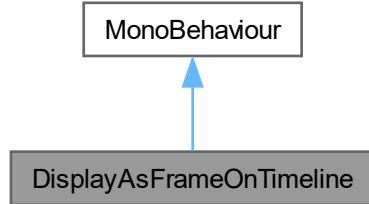
4.12 DisplayAsFrameOnTimeline Class Reference

This class is used for any UI element that should be displayed as a certain frame on the timeline for example a keyframe for frame 47 always needs to be positioned where frame 47 is on the timeline, regardless of scale / zoom level of the timeline. This class handles that.

Inheritance diagram for DisplayAsFrameOnTimeline:



Collaboration diagram for DisplayAsFrameOnTimeline:



Public Attributes

- long `Frame`

Protected Member Functions

- virtual void `Start ()`
- virtual void `Update ()`

Properties

- `Timeline Timeline [get, protected set]`
- `RectTransform Rect [get, protected set]`
- `CanvasGroup CanvasGroup [get, protected set]`
- `bool IsDraggable [get, protected set]`

Private Member Functions

- void `Hide ()`
- void `Show ()`
- void `RecalculateFrameBasedOnPosition ()`

4.12.1 Detailed Description

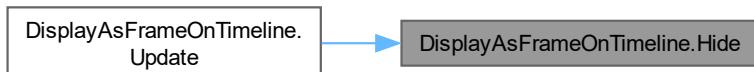
This classed is used for any UI element that should be displayed as a certain frame on the timeline for example a keyframe for frame 47 always needs to be positioned where frame 47 is on the timeline, regardless of scale / zoom level of the timeline. This class handles that.

4.12.2 Member Function Documentation

4.12.2.1 Hide()

```
void DisplayAsFrameOnTimeline.Hide ( ) [private]
```

Here is the caller graph for this function:



4.12.2.2 RecalculateFrameBasedOnPosition()

```
void DisplayAsFrameOnTimeline.RecalculateFrameBasedOnPosition ( ) [private]
```

Here is the caller graph for this function:



4.12.2.3 Show()

```
void DisplayAsFrameOnTimeline.Show ( ) [private]
```

Here is the caller graph for this function:



4.12.2.4 Start()

```
virtual void DisplayAsFrameOnTimeline.Start () [protected], [virtual]
```

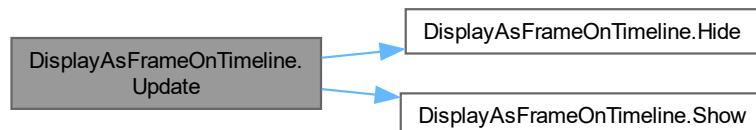
Here is the call graph for this function:



4.12.2.5 Update()

```
virtual void DisplayAsFrameOnTimeline.Update () [protected], [virtual]
```

Here is the call graph for this function:



4.12.3 Member Data Documentation

4.12.3.1 Frame

```
long DisplayAsFrameOnTimeline.Frame
```

4.12.4 Property Documentation

4.12.4.1 CanvasGroup

```
CanvasGroup DisplayAsFrameOnTimeline.CanvasGroup [get], [protected set]
```

4.12.4.2 IsDraggable

```
bool DisplayAsFrameOnTimeline.IsDraggable [get], [protected set]
```

4.12.4.3 Rect

```
RectTransform DisplayAsFrameOnTimeline.Rect [get], [protected set]
```

4.12.4.4 Timeline

```
Timeline DisplayAsFrameOnTimeline.Timeline [get], [protected set]
```

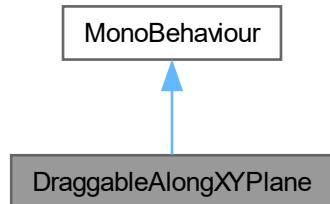
The documentation for this class was generated from the following file:

- Assets/Scripts/Video and Timeline/[DisplayAsFrameOnTimeline.cs](#)

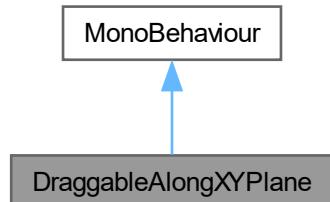
4.13 DraggableAlongXYPlane Class Reference

this script is attached to every AOI handle and allows the handle to be dragged along the XY plane of its parent(the AOI)

Inheritance diagram for DraggableAlongXYPlane:



Collaboration diagram for DraggableAlongXYPlane:



Public Member Functions

- void `OnMouseDown ()`
- void `OnMouseUp ()`

Public Attributes

- bool `FlipMesh` = false
- UnityEvent `MovementCompleted`
- UnityEvent `MovementStarted`

Protected Member Functions

- void `Move ()`

Protected Attributes

- Vector3 `_offset` = Vector3.zero

Properties

- bool `IsTryingToMove` = false [get, protected set]

Private Member Functions

- void `Update ()`
- void `OnDisable ()`

4.13.1 Detailed Description

this script is attached to every AOI handle and allows the handle to be dragged along the XY plane of its parent(the AOI)

4.13.2 Member Function Documentation

4.13.2.1 Move()

```
void DraggableAlongXYPlane.Move ( ) [protected]
```

Here is the caller graph for this function:



4.13.2.2 OnDisable()

```
void DraggableAlongXYPlane.OnDisable ( ) [private]
```

Here is the call graph for this function:



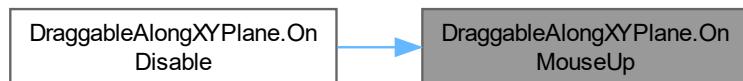
4.13.2.3 OnMouseDown()

```
void DraggableAlongXYPlane.OnMouseDown ( )
```

4.13.2.4 OnMouseUp()

```
void DraggableAlongXYPlane.OnMouseUp ( )
```

Here is the caller graph for this function:



4.13.2.5 Update()

```
void DraggableAlongXYPlane.Update ( ) [private]
```

Here is the call graph for this function:



4.13.3 Member Data Documentation

4.13.3.1 `_offset`

```
Vector3 DraggableAlongXYPlane._offset = Vector3.zero [protected]
```

4.13.3.2 `FlipMesh`

```
bool DraggableAlongXYPlane.FlipMesh = false
```

4.13.3.3 `MovementCompleted`

```
UnityEvent DraggableAlongXYPlane.MovementCompleted
```

4.13.3.4 `MovementStarted`

```
UnityEvent DraggableAlongXYPlane.MovementStarted
```

4.13.4 Property Documentation

4.13.4.1 `IsTryingToMove`

```
bool DraggableAlongXYPlane.IsTryingToMove = false [get], [protected set]
```

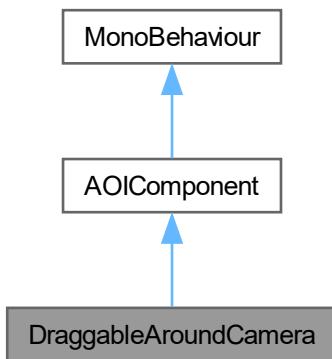
The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/AOI Handles/[DraggableAlongXYPlane.cs](#)

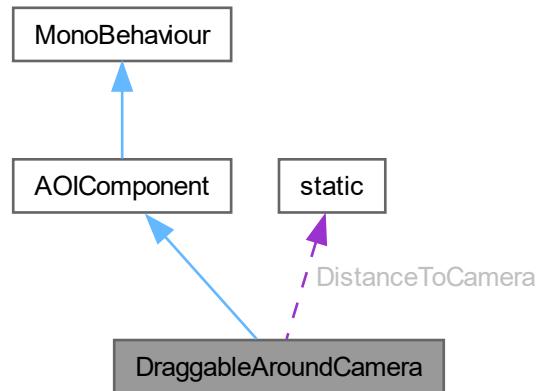
4.14 DraggableAroundCamera Class Reference

This script is attached to AOIs and allows them to be dragged around to position them within the 360 degree video.

Inheritance diagram for DraggableAroundCamera:



Collaboration diagram for DraggableAroundCamera:



Public Member Functions

- void [OnMouseDown \(\)](#)
Callback for when AOI is being clicked. If AOI is currently selected AOI start movement.
- void [OnMouseUp \(\)](#)
callback for when mouse button is released.

Public Attributes

- UnityEvent [MovementCompleted](#) = new()
- UnityEvent [MovementStarted](#) = new()

Static Public Attributes

- static float [DistanceToCamera](#) = 10

Protected Member Functions

- override void [OnActivate \(\)](#)
When AOI is being activated (due to click on aoi, due to click on list entry or right after spawn) check if cursor is above AOI, if it is, immediately allow movement instead of having to click again.
- virtual void [Move \(\)](#)
Calculates new AOI position based on cursor movement. AOI follows cursor around world-center.
- override void [OnDeactivate \(\)](#)
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.

Protected Member Functions inherited from [AOIComponent](#)

- virtual void [Start \(\)](#)
- virtual void [ActiveAOIchanged](#) (GameObject newAoi)

Callback for AOIManagers NewActiveAOI-Event. Called whenever user clicks on new AOI.
- abstract void [OnActivate \(\)](#)

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.
- abstract void [OnDeactivate \(\)](#)

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.
- virtual void [OnDestroy \(\)](#)

Protected Attributes

- Vector3 [_offset](#) = Vector3.zero

Properties

- bool [IsTryingToMove](#) = false [get, protected set]
- bool [IsTryingToMoveDueToBeingActivited](#) = false [get, protected set]

Private Member Functions

- void [Update \(\)](#)

4.14.1 Detailed Description

This script is attached to AOIs and allows them to be dragged around to position them within the 360 degree video.

4.14.2 Member Function Documentation

4.14.2.1 Move()

```
virtual void DraggableAroundCamera.Move ( ) [protected], [virtual]
```

Calculates new AOI position based on cursor movement. AOI follows cursor around world-center.

Here is the caller graph for this function:



4.14.2.2 OnActivate()

```
override void DraggableAroundCamera.OnActivate () [protected], [virtual]
```

When AOI is being activated (due to click on aoi, due to click on list entry or right after spawn) check if cursor is above AOI, if it is, immediately allow movement instead of having to click again.

Implements [AOIComponent](#).

Here is the call graph for this function:



4.14.2.3 OnDeactivate()

```
override void DraggableAroundCamera.OnDeactivate () [protected], [virtual]
```

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.

Implements [AOIComponent](#).

4.14.2.4 OnMouseDown()

```
void DraggableAroundCamera.OnMouseDown ()
```

Callback for when AOI is being clicked. If AOI is currently selected AOI start movement.

Here is the call graph for this function:



Here is the caller graph for this function:



4.14.2.5 OnMouseUp()

```
void DraggableAroundCamera.OnMouseUp ( )
```

callback for when mouse button is released.

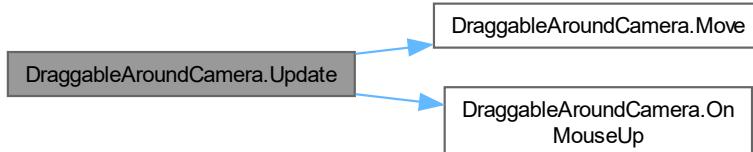
Here is the caller graph for this function:



4.14.2.6 Update()

```
void DraggableAroundCamera.Update ( ) [private]
```

Here is the call graph for this function:



4.14.3 Member Data Documentation

4.14.3.1 _offset

```
Vector3 DraggableAroundCamera._offset = Vector3.zero [protected]
```

4.14.3.2 DistanceToCamera

```
float DraggableAroundCamera.DistanceToCamera = 10f [static]
```

4.14.3.3 MovementCompleted

```
UnityEvent DraggableAroundCamera.MovementCompleted = new()
```

4.14.3.4 MovementStarted

```
UnityEvent DraggableAroundCamera.MovementStarted = new()
```

4.14.4 Property Documentation

4.14.4.1 IsTryingToMove

```
bool DraggableAroundCamera.IsTryingToMove = false [get], [protected set]
```

4.14.4.2 IsTryingToMoveDueToBeingActivtved

```
bool DraggableAroundCamera.IsTryingToMoveDueToBeingActivtved = false [get], [protected set]
```

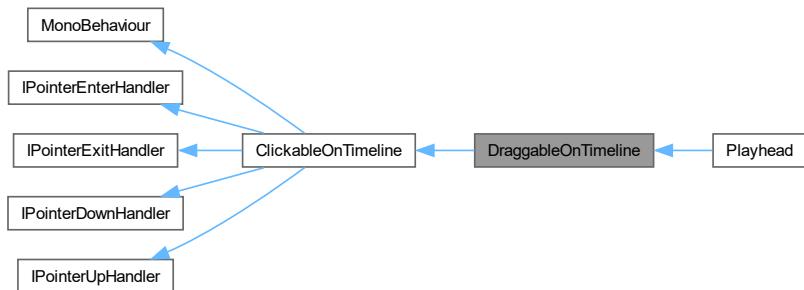
The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/AOI Components/[DraggableAroundCamera.cs](#)

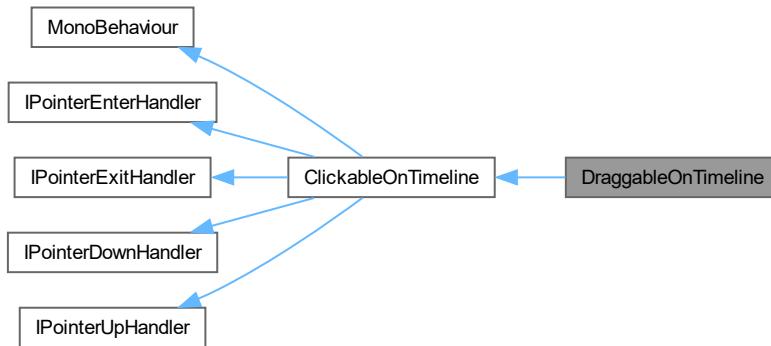
4.15 DraggableOnTimeline Class Reference

This class allows an object which is displayed on the timeline (using `DisplayAsFrameOnTimelineCOnponent`) to be dragged along the timeline using the mouse. (Used by the playhead to scrub along the timeline)

Inheritance diagram for DraggableOnTimeline:



Collaboration diagram for DraggableOnTimeline:



Public Member Functions

- delegate void [MovementCompleted \(\)](#)
- override void [OnPointerDown \(PointerEventData eventData\)](#)
If an item on the timeline has been clicked it marks itself as clicked and takes the hover away from the timeline.
- virtual void [SetAnchoredX \(float new_x\)](#)

Public Member Functions inherited from [ClickableOnTimeline](#)

- virtual void [OnPointerEnter \(PointerEventData eventData\)](#)
When the cursor hover above an item on the timeline the item marks itself as hovered.
- virtual void [OnPointerExit \(PointerEventData eventData\)](#)
If the cursor leaves the area of the clickable item but has previously been clicked and the mouse is still pressed down, the item makes sure that the timeline still doesn't register as being hovered, because the user is still interacting with the item itself, since the mouse is still pressed down (the user might e.g. be trying to drag an item across the timeline.). If the cursor leaves the area of the item and the item has not been clicked, it sets itself as not hovered and hands over the hover to the timeline.
- virtual void [OnPointerDown \(PointerEventData eventData\)](#)
If an item on the timeline has been clicked it marks itself as clicked and takes the hover away from the timeline.
- virtual void [OnPointerUp \(PointerEventData eventData\)](#)
On Pointer up the item registers itself as not clicked anymore, allows edge snapping of the timeline abd handover hover to the timeline.

Public Attributes

- [MovementCompleted MovementCompletedSubscribers](#)

Protected Member Functions

- override void [Start \(\)](#)
- override void [Update \(\)](#)

After a clickable item on the timeline has been clicked, for as long as the mouse is still down, the update function makes sure that the timeline never registers as being "hovered", since the user is hovering and item on the timeline, not the timeline itself.

- virtual void [OnPositionChange \(\)](#)
- virtual void [OnMovementCompleted \(\)](#)
- virtual bool [MovementCondition \(\)](#)

- virtual void [Start \(\)](#)
- virtual void [Update \(\)](#)

After a clickable item on the timeline has been clicked, for as long as the mouse is still down, the update function makes sure that the timeline never registers as being "hovered", since the user is hovering and item on the timeline, not the timeline itself.

Protected Attributes

- Vector2 [_mouseOffset](#)

Protected Attributes inherited from [ClickableOnTimeline](#)

- RectTransform [_rect](#)

Properties

- Canvas [Canvas](#) [get, protected set]
- bool [IsBeingDragged = false](#) [get, protected set]

Properties inherited from [ClickableOnTimeline](#)

- Timeline [Timeline](#) [get, protected set]
- bool [IsHovered = false](#) [get, protected set]
- bool [IsClickedAndMouseStillDown = false](#) [get, protected set]

4.15.1 Detailed Description

This class allows an object which is displayed on the timeline (using `DisplayAsFrameOnTimelineCOnponent`) to be dragged along the timeline using the mouse. (Used by the playhead to scrub along the timeline)

4.15.2 Member Function Documentation

4.15.2.1 MovementCompleted()

```
delegate void DraggableOnTimeline.MovementCompleted ( )
```

4.15.2.2 MovementCondition()

```
virtual bool DraggableOnTimeline.MovementCondition ( ) [protected], [virtual]
```

Reimplemented in [Playhead](#).

Here is the caller graph for this function:



4.15.2.3 OnMovementCompleted()

```
virtual void DraggableOnTimeline.OnMovementCompleted ( ) [protected], [virtual]
```

Reimplemented in [Playhead](#).

Here is the caller graph for this function:



4.15.2.4 OnPointerDown()

```
override void DraggableOnTimeline.OnPointerDown ( PointerEventData eventData ) [virtual]
```

If an item on the timeline has been clicked it marks itself as clicked and takes the hover away from the timeline.

Parameters

eventData	<input type="text"/>
-----------	----------------------

Reimplemented from [ClickableOnTimeline](#).

4.15.2.5 OnPositionChange()

```
virtual void DraggableOnTimeline.OnPositionChange ( ) [protected], [virtual]
```

Reimplemented in [Playhead](#).

Here is the caller graph for this function:



4.15.2.6 SetAnchoredX()

```
virtual void DraggableOnTimeline.SetAnchoredX (
    float new_x ) [virtual]
```

Reimplemented in [Playhead](#).

4.15.2.7 Start()

```
override void DraggableOnTimeline.Start ( ) [protected], [virtual]
```

Reimplemented from [ClickableOnTimeline](#).

4.15.2.8 Update()

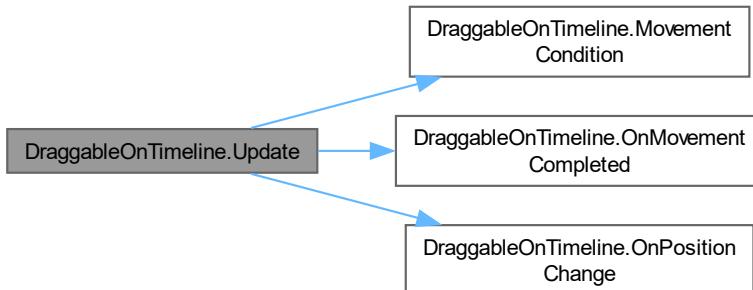
```
override void DraggableOnTimeline.Update ( ) [protected], [virtual]
```

After a clickable item on the timeline has been clicked, for as long as the mouse is still down, the update function makes sure that the timeline never registers as being "hovered", since the user is hovering an item on the timeline, not the timeline itself.

Reimplemented from [ClickableOnTimeline](#).

Reimplemented in [Playhead](#).

Here is the call graph for this function:



4.15.3 Member Data Documentation

4.15.3.1 `_mouseOffset`

```
Vector2 DraggableOnTimeline._mouseOffset [protected]
```

4.15.3.2 `MovementCompletedSubscribers`

```
MovementCompleted DraggableOnTimeline.MovementCompletedSubscribers
```

4.15.4 Property Documentation

4.15.4.1 `Canvas`

```
Canvas DraggableOnTimeline.Canvas [get], [protected set]
```

4.15.4.2 `IsBeingDragged`

```
bool DraggableOnTimeline.IsBeingDragged = false [get], [protected set]
```

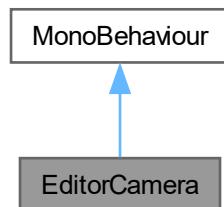
The documentation for this class was generated from the following file:

- Assets/Scripts/Video and Timeline/[DraggableOnTimeline.cs](#)

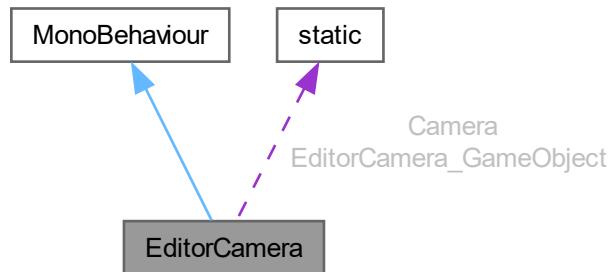
4.16 EditorCamera Class Reference

This script is attached to the editor camera and does two things a) gives a static reference to the [EditorCamera](#) to be used for e.g. position calculation etc. b) matches the editor camera rotation to the VR cameras rotation if flag is set by user.

Inheritance diagram for EditorCamera:



Collaboration diagram for EditorCamera:



Static Public Attributes

- static GameObject [EditorCamera_GameObject](#)
- static Camera [Camera](#)

Private Member Functions

- void [Start \(\)](#)
- void [OnVrStatusChange \(VRManager.VRStatus status\)](#)
Callback for VRManager VRStatusChange-Event. When VR is being enabled and TestScene is enabled AOIs and 360 degree video are hidden from view.
- void [OnTestSceneToggled \(bool isTestSceneEnabled\)](#)
Callback for when user is toggling testscene from VRSettings Menu.
- void [OnRecordingToggled \(bool is_recording\)](#)
Callback for EyeRecorders recording event. When recording is started, make sure that skybox with 360degree video as well as AOIs are visible.
- void [Update \(\)](#)
Update function. If tracking of VR camera is enabled the editor camera updates its rotation every frame to match the VR Camera.

4.16.1 Detailed Description

This script is attached to the editor camera and does two things a) gives a static reference to the [EditorCamera](#) to be used for e.g. position calculation etc. b) matches the editor camera rotation to the VR cameras rotation if flag is set by user.

4.16.2 Member Function Documentation

4.16.2.1 OnRecordingToggled()

```
void EditorCamera.OnRecordingToggled (
    bool is_recording ) [private]
```

Callback for EyeRecorders recording event. When recording is started, make sure that skybox with 360degree video as well as AOIs are visible.

Parameters

<i>is_recording</i>	bool if recording has been turned on or off.
---------------------	--

Here is the caller graph for this function:



4.16.2.2 OnTestSceneToggled()

```
void EditorCamera.OnTestSceneToggled (
    bool isTestSceneEnabled ) [private]
```

Callback for when user is toggling testscene from VRSettings Menu.

Parameters

<i>isTestSceneEnabled</i>	bool whether or not testscene is enabled
---------------------------	--

Here is the caller graph for this function:



4.16.2.3 OnVrStatusChange()

```
void EditorCamera.OnVrStatusChange (
    VRManager::VRStatus status ) [private]
```

Callback for [VRManager](#) VRStatusChange-Event. When VR is being enabled and TestScene is enabled AOIs and 360 degree video are hidden from view.

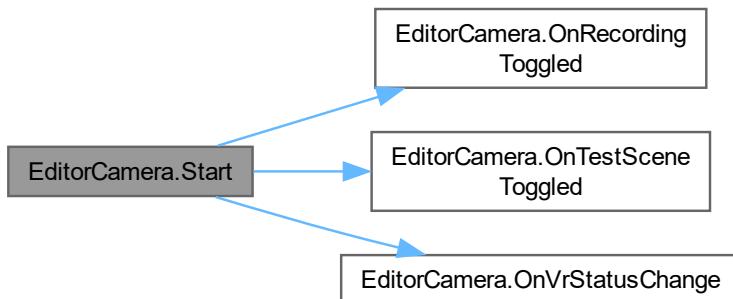
Parameters`status`

Here is the caller graph for this function:

**4.16.2.4 Start()**

```
void EditorCamera.Start ( ) [private]
```

Here is the call graph for this function:

**4.16.2.5 Update()**

```
void EditorCamera.Update ( ) [private]
```

Update function. If tracking of VR camera is enabled the editor camera updates its rotation every frame to match the VR Camera.

4.16.3 Member Data Documentation**4.16.3.1 Camera**

```
Camera EditorCamera.Camera [static]
```

4.16.3.2 EditorCamera_GameObject

```
GameObject EditorCamera.EditorCamera_GameObject [static]
```

The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/[EditorCamera.cs](#)

4.17 EventDetection.Event Class Reference

Public Types

- enum [EventType](#) { [Dwell](#) , [Fixation](#) }

Public Member Functions

- [Event](#) ([EventType](#) type, int duration, int startTimestamp, int endTimestamp=-1, string aoiName="null")

Public Attributes

- [EventType](#) Type
- int Duration
- int StartTimestamp
- int EndTimestamp
- string AoiName

4.17.1 Member Enumeration Documentation

4.17.1.1 EventType

```
enum EventDetection.Event.EventType
```

Enumerator

Dwell	
Fixation	

4.17.2 Constructor & Destructor Documentation

4.17.2.1 Event()

```
EventDetection.Event.Event (
    EventType type,
    int duration,
```

```
int startTimestamp,  
int endTimestamp = -1,  
string aoiName = "null" )
```

4.17.3 Member Data Documentation

4.17.3.1 AoiName

```
string EventDetection.Event.AoiName
```

4.17.3.2 Duration

```
int EventDetection.Event.Duration
```

4.17.3.3 EndTimestamp

```
int EventDetection.Event.EndTimestamp
```

4.17.3.4 StartTimestamp

```
int EventDetection.Event.StartTimestamp
```

4.17.3.5 Type

```
EventType EventDetection.Event.Type
```

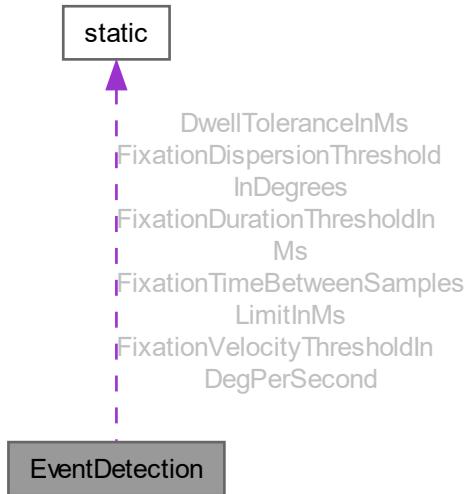
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/[EventDetection.cs](#)

4.18 EventDetection Class Reference

This script contains all the event-detection algorithms for detecting dwells and fixations with their respective parameters like duration etc.

Collaboration diagram for EventDetection:



Classes

- class [AoiParameters](#)
- class [Event](#)

Public Member Functions

- `EventDetection (List< EyeTrackingDataSample > rawDataSamples)`
- `void FilterInvalidData ()`
Filters all datasamples that don't have a valid gaze direction.
- `int FindIndexOfNextValidSample (int startIndex, bool lookAhead, ulong validity_flag)`
helper function that finds next valid data sample closest to specified start index. Not used anymore.
- `bool DetectDwells ()`
Algorithm for detecting dwells.
- `bool DetectFixations ()`
I-DVT algorithm that detects fixations. Based on lanes-Jurado et al. (2020) <https://pubmed.ncbi.nlm.nih.gov/32883026/>.
- `float EventDurationProportion (string aoi_name, Event.EventType eventType)`
Calculates the proportion of duration for dwells or fixations for a given AOI.
- `float EventCountProportion (string aoi_name, Event.EventType eventType)`
Calculates the proportion of occurrences for dwells or fixations for a given AOI.
- `List< string > UniqueAoiNames ()`
Little helper that returns a list of the unique AOIs that are found in the filtered data.

Public Attributes

- `List< EyeTrackingDataSample > RawDataSamples`
- `List< EyeTrackingDataSample > FilteredDataSamples`
- `bool[] Fixationsample_flags`
- `IDictionary< string, AoiParameters > Results = new Dictionary<string, AoiParameters>()`

Static Public Attributes

- static int `FixationDurationThresholdInMs` = 150
- static int `FixationTimeBetweenSamplesLimitInMs` = 32
- static float `FixationDispersionThresholdInDegrees` = 1
- static float `FixationVelocityThresholdInDegPerSecond` = 35
- static int `DwellToleranceInMs` = 100

Properties

- List< `Event` > `Dwells` [get, private set]
- List< `Event` > `Fixations` [get, private set]

Private Member Functions

- int `GetSampleIndexAtMinFixationDuration` (List< `EyeTrackingDataSample` > samples, int startIndex=0)

Helper Function that takes a start index and outputs the index if the first data sample that is FixationDurationThresholdInMs milliseconds away from start index. It basically spans a window of 150ms (or whatever value the user chooses) to be used in the fixation detection.

4.18.1 Detailed Description

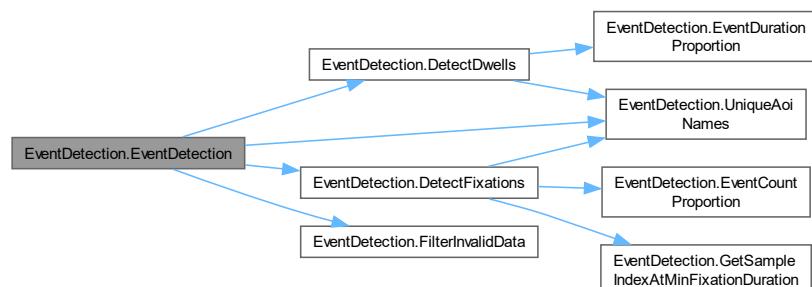
This script contains all the event-detection algorithms for detecting dwells and fixations with their respective parameters like duration etc.

4.18.2 Constructor & Destructor Documentation

4.18.2.1 EventDetection()

```
EventDetection.EventDetection (
    List< EyeTrackingDataSample > rawDataSamples )
```

Here is the call graph for this function:



4.18.3 Member Function Documentation

4.18.3.1 DetectDwells()

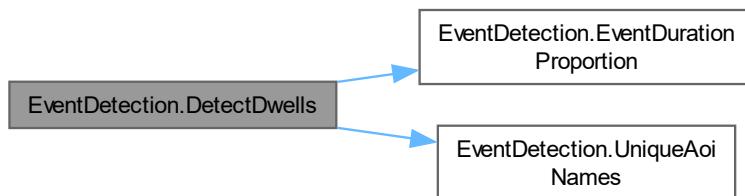
```
bool EventDetection.DetectDwells ( )
```

Algorithm for detecting dwells.

Returns

Returns true if detection had valid data samples to work with.

Here is the call graph for this function:



Here is the caller graph for this function:



4.18.3.2 DetectFixations()

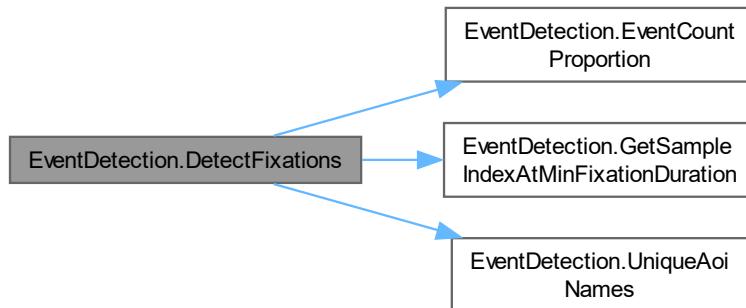
```
bool EventDetection.DetectFixations ( )
```

I-DVT algorithm that detects fixations. Based on Ianes-Jurado et al. (2020) <https://pubmed.ncbi.nlm.nih.gov/32883026/>.

Returns

Returns true when fixation detection was able to run without errors.

Here is the call graph for this function:



Here is the caller graph for this function:

**4.18.3.3 EventCountProportion()**

```

float EventDetection.EventCountProportion (
    string aoi_name,
    Event::EventType eventType )
  
```

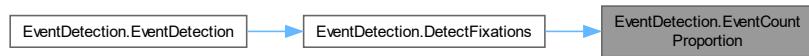
Calculates the proportion of occurrences for dwells or fixations for a given AOI.

Parameters

<i>aoi_name</i>	Name of AOI for which to calculate occurrences-proportion
<i>eventType</i>	Dwells or Fixations

Returns

Here is the caller graph for this function:

**4.18.3.4 EventDurationProportion()**

```
float EventDetection.EventDurationProportion (
    string aoi_name,
    Event::EventType eventType )
```

Calculates the proportion of duration for dwells or fixations for a given AOI.

Parameters

<i>aoi_name</i>	Name of AOI for which to calculate duration proportion
<i>eventType</i>	Dwells or Fixations

Returns

Here is the caller graph for this function:

**4.18.3.5 FilterInvalidData()**

```
void EventDetection.FilterInvalidData ( )
```

Filters all datasamples that don't have a valid gaze direction.

Here is the caller graph for this function:



4.18.3.6 FindIndexOfNextValidSample()

```
int EventDetection.FindIndexOfNextValidSample (
    int startIndex,
    bool lookAhead,
    ulong validity_flag )
```

helper function that finds next valid data sample closest to specified start index. Not used anymore.

Parameters

<i>startIndex</i>	
<i>lookAhead</i>	if true, looks right of start index, if false looks left
<i>validity_flag</i>	flag in bitmask to be checked

Returns

4.18.3.7 GetSampleIndexAtMinFixationDuration()

```
int EventDetection.GetSampleIndexAtMinFixationDuration (
    List< EyeTrackingDataSample > samples,
    int startIndex = 0 ) [private]
```

Helper Function that takes a start index and outputs the index if the first data sample that is FixationDurationThresholdInMs milliseconds away from start index. It basically spans a window of 150ms (or whatever value the user chooses) to be used in the fixation detection.

Parameters

<i>samples</i>	Samples to check
<i>startIndex</i>	Start index to check from

Returns

Index at position of FixationDurationThresholdInMs away from start index

Here is the caller graph for this function:



4.18.3.8 UniqueAoiNames()

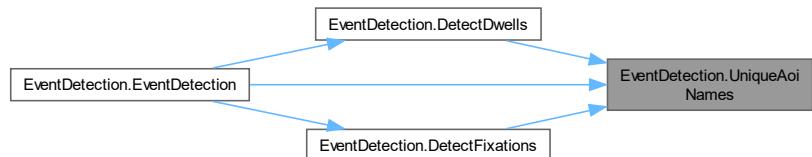
```
List< string > EventDetection.UniqueAoiNames( )
```

Little helper that returns a list of the unique AOIs that are found in the filtered data.

Returns

List of unique AOI names.

Here is the caller graph for this function:



4.18.4 Member Data Documentation

4.18.4.1 DwellToleranceInMs

```
int EventDetection.DwellToleranceInMs = 100 [static]
```

4.18.4.2 FilteredDataSamples

```
List<EyeTrackingDataSample> EventDetection.FilteredDataSamples
```

4.18.4.3 FixationDispersionThresholdInDegrees

```
float EventDetection.FixationDispersionThresholdInDegrees = 1 [static]
```

4.18.4.4 FixationDurationThresholdInMs

```
int EventDetection.FixationDurationThresholdInMs = 150 [static]
```

4.18.4.5 Fixationsample_flags

```
bool [] EventDetection.Fixationsample_flags
```

4.18.4.6 FixationTimeBetweenSamplesLimitInMs

```
int EventDetection.FixationTimeBetweenSamplesLimitInMs = 32 [static]
```

4.18.4.7 FixationVelocityThresholdInDegPerSecond

```
float EventDetection.FixationVelocityThresholdInDegPerSecond = 35 [static]
```

4.18.4.8 RawDataSamples

```
List<EyeTrackingDataSample> EventDetection.RawDataSamples
```

4.18.4.9 Results

```
IDictionary<string, AoiParameters> EventDetection.Results = new Dictionary<string, AoiParameters>()
```

4.18.5 Property Documentation

4.18.5.1 Dwells

```
List<Event> EventDetection.Dwells [get], [private set]
```

4.18.5.2 Fixations

```
List<Event> EventDetection.Fixations [get], [private set]
```

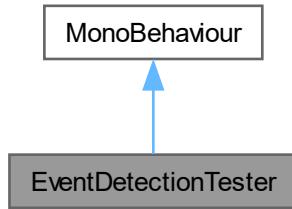
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/[EventDetection.cs](#)

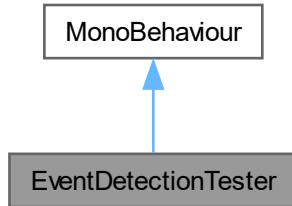
4.19 EventDetectionTester Class Reference

Some testing code for the event detection. Generates a list EyeTrackingDataSamples, then creates an event detection object and passes the list as parameter. The created csv file can be checked manually and compared to the created EyeTrackingDataSamples.

Inheritance diagram for EventDetectionTester:



Collaboration diagram for EventDetectionTester:



Classes

- class [TesterSample](#)

Static Public Member Functions

- static `IDictionary< string, EventDetection.AoiParameters > RunTest (bool logResults=true)`

Private Member Functions

- void `Start ()`

4.19.1 Detailed Description

Some testing code for the event detection. Generates a list EyeTrackingDataSamples, then creates an event detection object and passes the list as parameter. The created csv file can be checked manually and compared to the created EyeTrackingDataSamples.

4.19.2 Member Function Documentation

4.19.2.1 RunTest()

```
static IDictionary< string, EventDetection.AoiParameters > EventDetectionTester.RunTest ( 
    bool logResults = true ) [static]
```

Here is the caller graph for this function:



4.19.2.2 Start()

```
void EventDetectionTester.Start ( ) [private]
```

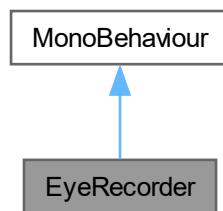
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/Simulation/EventDetectionTester.cs

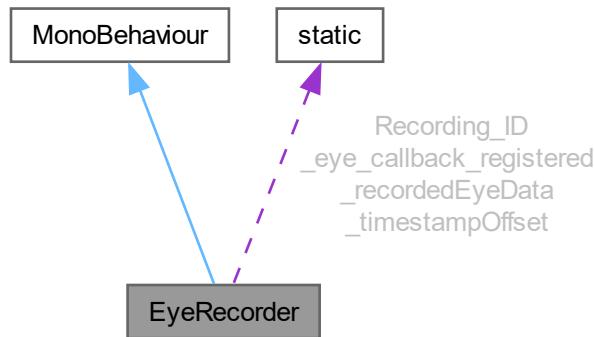
4.20 EyeRecorder Class Reference

Derived from SRanipal example script. This script is responsible for getting the eyedata from the VR-Headset, storing the data during recording as well as carrying out the raycast to check whether or not the gaze is hitting a AOI. Note: Callback runs on a separate thread to report at ~120hz. Unity is not threadsafe and cannot call any UnityEngine api from within callback thread.

Inheritance diagram for EyeRecorder:



Collaboration diagram for EyeRecorder:



Classes

- class **MonoPInvokeCallbackAttribute**

Required class for IL2CPP scripting backend support.

Public Member Functions

- void **ToggleRecording ()**
Callback for Recording button at top right corner of UI. Toggles recording on or off.
- void **StartRecording ()**
Starts the recording by mainly invoking the RecordData method with InvokeRepeating.
- void **FinishRecording ()**
stops the recording, creates event detection and logs data. starts python process for visualizing data.

Public Attributes

- double **Sensitivity** = 1

Static Public Attributes

- static string **Recording_ID**

Properties

- static **EyeRecorder Instance** [get, private set]
- static bool **IsRecording** = false [get, private set]
- static UnityEvent< bool > **OnRecordingToggled** = new() [get, private set]
- static EyeData **CurrentEyeData** = new EyeData() [get, private set]
- static EyeData **PreviousEyeData** = new EyeData() [get, private set]
- static List< EyeTrackingDataSample > **RawData** = new List< EyeTrackingDataSample>() [get, private set]

Private Member Functions

- void [SetIsRecordingAndNotify](#) (bool new_status)
- void [Start](#) ()
- void [Update](#) ()
- void [OnDisable](#) ()
- void [OnApplicationQuit](#) ()

Static Private Member Functions

- static void [Release](#) ()

Release callback thread when disabled or quit.
- static void [EyeCallback](#) (ref EyeData eye_data)

Eye tracking data callback thread. Reports data at ~120hz MonoPInvokeCallback attribute required for IL2CPP scripting backend.

Static Private Attributes

- static bool [_eye_callback_registered](#) = false
- static int [_timestampOffset](#) = 0
- static List< EyeData > [_recordedEyeData](#) = new List<EyeData>()

4.20.1 Detailed Description

Derived from SRanipal example script. This script is responsible for getting the eyedata from the VR-Headset, storing the data during recording as well as carrying out the raycast to check whether or not the gaze is hitting a AOI. Note: Callback runs on a separate thread to report at ~120hz. Unity is not threadsafe and cannot call any UnityEngine api from within callback thread.

4.20.2 Member Function Documentation

4.20.2.1 EyeCallback()

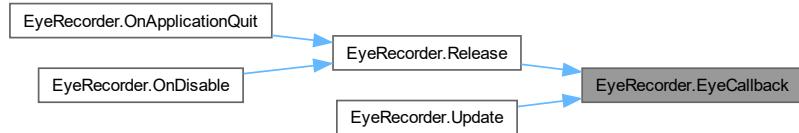
```
static void EyeRecorder.EyeCallback (
    ref EyeData eye_data ) [static], [private]
```

Eye tracking data callback thread. Reports data at ~120hz MonoPInvokeCallback attribute required for IL2CPP scripting backend.

Parameters

<code>eye_data</code>	Reference to latest eye_data
-----------------------	------------------------------

Here is the caller graph for this function:

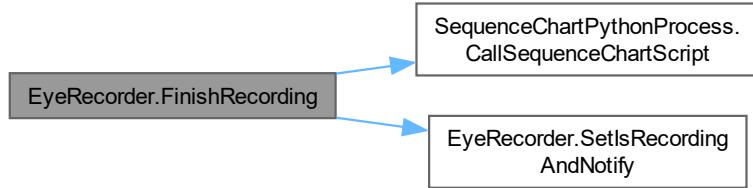


4.20.2.2 FinishRecording()

```
void EyeRecorder.FinishRecording( )
```

stops the recording, creates event detection and logs data. starts python process for visualizing data.

Here is the call graph for this function:



Here is the caller graph for this function:



4.20.2.3 OnApplicationQuit()

```
void EyeRecorder.OnApplicationQuit( ) [private]
```

Here is the call graph for this function:



4.20.2.4 OnDisable()

```
void EyeRecorder.OnDisable ( ) [private]
```

Here is the call graph for this function:



4.20.2.5 Release()

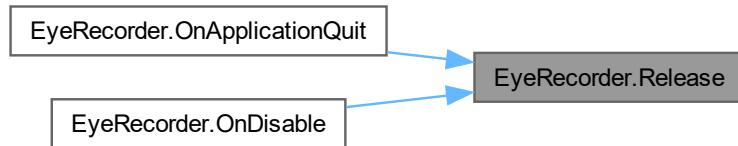
```
static void EyeRecorder.Release ( ) [static], [private]
```

Release callback thread when disabled or quit.

Here is the call graph for this function:



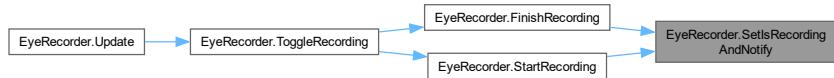
Here is the caller graph for this function:



4.20.2.6 SetIsRecordingAndNotify()

```
void EyeRecorder.SetIsRecordingAndNotify ( bool new_status ) [private]
```

Here is the caller graph for this function:



4.20.2.7 Start()

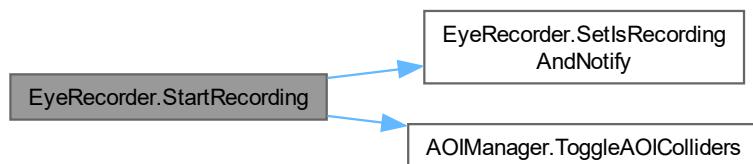
```
void EyeRecorder.Start ( ) [private]
```

4.20.2.8 StartRecording()

```
void EyeRecorder.StartRecording ( )
```

Starts the recording by mainly invoking the RecordData method with InvokeRepeating.

Here is the call graph for this function:



Here is the caller graph for this function:

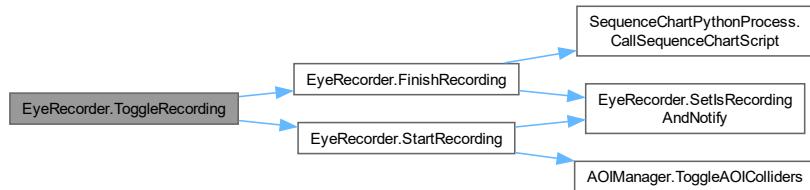


4.20.2.9 ToggleRecording()

```
void EyeRecorder.ToggleRecording( )
```

Callback for Recording button at top right corner of UI. Toggles recording on or off.

Here is the call graph for this function:



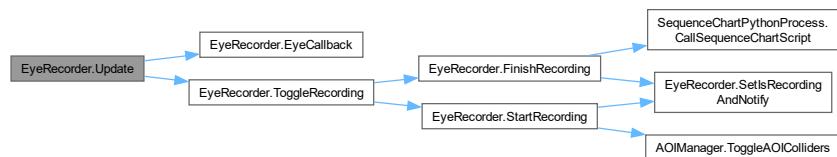
Here is the caller graph for this function:



4.20.2.10 Update()

```
void EyeRecorder.Update( ) [private]
```

Here is the call graph for this function:



4.20.3 Member Data Documentation

4.20.3.1 _eye_callback_registered

```
bool EyeRecorder._eye_callback_registered = false [static], [private]
```

4.20.3.2 `_recordedEyeData`

```
List<EyeData> EyeRecorder._recordedEyeData = new List<EyeData>() [static], [private]
```

4.20.3.3 `_timestampOffset`

```
int EyeRecorder._timestampOffset = 0 [static], [private]
```

4.20.3.4 `Recording_ID`

```
string EyeRecorder.Recording_ID [static]
```

4.20.3.5 `Sensitivity`

```
double EyeRecorder.Sensitivity = 1
```

4.20.4 Property Documentation

4.20.4.1 `CurrentEyeData`

```
EyeData EyeRecorder.CurrentEyeData = new EyeData() [static], [get], [private set]
```

4.20.4.2 `Instance`

```
EyeRecorder EyeRecorder.Instance [static], [get], [private set]
```

4.20.4.3 `IsRecording`

```
bool EyeRecorder.IsRecording = false [static], [get], [private set]
```

4.20.4.4 `OnRecordingToggled`

```
UnityEvent<bool> EyeRecorder.OnRecordingToggled = new() [static], [get], [private set]
```

4.20.4.5 `PreviousEyeData`

```
EyeData EyeRecorder.PreviousEyeData = new EyeData() [static], [get], [private set]
```

4.20.4.6 RawData

```
List<EyeTrackingDataSample> EyeRecorder.RawData = new List<EyeTrackingDataSample>() [static],  
[get], [private set]
```

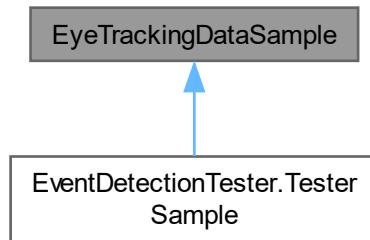
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/[EyeRecorder.cs](#)

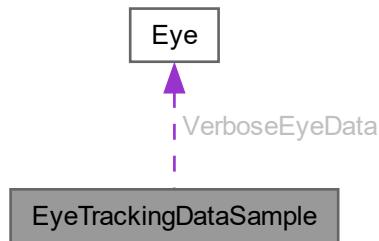
4.21 EyeTrackingDataSample Class Reference

Dataclass for storing all relevant data for a single eyetracking sample.

Inheritance diagram for EyeTrackingDataSample:



Collaboration diagram for EyeTrackingDataSample:



Public Member Functions

- [EyeTrackingDataSample](#) (int timestamp, long frame, Vector3 direction, ViveSR.anipal.Eye.VerboseData verboseData, string aoiName="null")
- [EyeTrackingDataSample](#) ()

Public Attributes

- int [Timestamp](#)
- long [VideoFrame](#)
- Vector3 [GazeDirection](#)
- string [AoiName](#)
- ViveSR.anipal.Eye.VerboseData [VerboseEyeData](#)

4.21.1 Detailed Description

Dataclass for storing all relevant data for a single eyetracking sample.

4.21.2 Constructor & Destructor Documentation

4.21.2.1 EyeTrackingDataSample() [1/2]

```
EyeTrackingDataSample.EyeTrackingDataSample (
    int timestamp,
    long frame,
    Vector3 direction,
    ViveSR::anipal::Eye::VerboseData verboseData,
    string aoiName = "null" )
```

4.21.2.2 EyeTrackingDataSample() [2/2]

```
EyeTrackingDataSample.EyeTrackingDataSample ( )
```

4.21.3 Member Data Documentation

4.21.3.1 AoiName

```
string EyeTrackingDataSample.AoiName
```

4.21.3.2 GazeDirection

```
Vector3 EyeTrackingDataSample.GazeDirection
```

4.21.3.3 Timestamp

```
int EyeTrackingDataSample.Timestamp
```

4.21.3.4 VerboseEyeData

```
ViveSR.anipal.Eye.VerboseData EyeTrackingDataSample.VerboseEyeData
```

4.21.3.5 VideoFrame

```
long EyeTrackingDataSample.VideoFrame
```

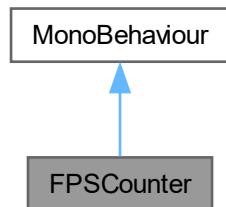
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/[EyeTrackingDataSample.cs](#)

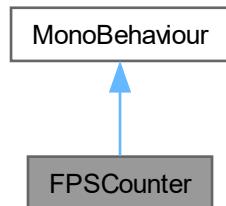
4.22 FPSCOUNTER Class Reference

Script to display FPS counter on the top right of screen. Source: <https://forum.unity.com/threads/fps-counter.505495/>.

Inheritance diagram for FPSCOUNTER:



Collaboration diagram for FPSCOUNTER:



Public Attributes

- TMPro.TMP_Text Text

Private Member Functions

- void `Awake()`
- void `Update()`

Private Attributes

- Dictionary< int, string > `CachedNumberStrings` = new()
- int[] `_frameRateSamples`
- int `_cacheNumbersAmount` = 300
- int `_averageFromAmount` = 30
- int `_averageCounter` = 0
- int `_currentAveraged`

4.22.1 Detailed Description

Script to display FPS counter on the top right of screen. Source: <https://forum.unity.com/threads/fps-counter.505495/>.

4.22.2 Member Function Documentation

4.22.2.1 `Awake()`

```
void FPSCOUNTER.Awake ( ) [private]
```

4.22.2.2 `Update()`

```
void FPSCOUNTER.Update ( ) [private]
```

4.22.3 Member Data Documentation

4.22.3.1 `_averageCounter`

```
int FPSCOUNTER._averageCounter = 0 [private]
```

4.22.3.2 `_averageFromAmount`

```
int FPSCOUNTER._averageFromAmount = 30 [private]
```

4.22.3.3 `_cacheNumbersAmount`

```
int FPSCOUNTER._cacheNumbersAmount = 300 [private]
```

4.22.3.4 `_currentAveraged`

```
int FPSCounter._currentAveraged [private]
```

4.22.3.5 `_frameRateSamples`

```
int [] FPSCounter._frameRateSamples [private]
```

4.22.3.6 `CachedNumberStrings`

```
Dictionary<int, string> FPSCounter.CachedNumberStrings = new() [private]
```

4.22.3.7 `Text`

```
TMP TMP_Text FPSCounter.Text
```

The documentation for this class was generated from the following file:

- Assets/Scripts/Application/[FPSCounter.cs](#)

4.23 Animation.FrameData Class Reference

Public Attributes

- GameObject [UIrepresentation](#)
- long [VideoFrame](#)
- Vector3[] [Vertices](#)
- Vector3 [Position](#)
- Quaternion [Rotation](#)
- bool [Visibility](#) = true

4.23.1 Member Data Documentation

4.23.1.1 `Position`

```
Vector3 Animation.FrameData.Position
```

4.23.1.2 `Rotation`

```
Quaternion Animation.FrameData.Rotation
```

4.23.1.3 `UIrepresentation`

```
GameObject Animation.FrameData.UIrepresentation
```

4.23.1.4 Vertices

```
Vector3 [] Animation.FrameData.Vertices
```

4.23.1.5 VideoFrame

```
long Animation.FrameData.VideoFrame
```

4.23.1.6 Visibility

```
bool Animation.FrameData.Visibility = true
```

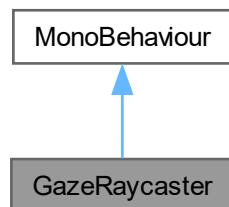
The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/AOI Components/[Animation.cs](#)

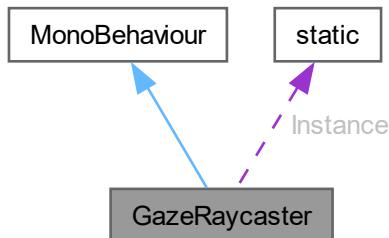
4.24 GazeRaycaster Class Reference

Creates a raycast from current eyedata gaze direction. If an object with the [RaycastHitHandler](#) component is hit by racast this script calls the OnRaycastEnter and OnRaycastExit methods on entry and exit of the object. This allows objects to react to beeing looked at by the participant.

Inheritance diagram for GazeRaycaster:



Collaboration diagram for GazeRaycaster:



Static Public Attributes

- static [GazeRaycaster Instance](#)

Properties

- GameObject [CurrentlyHitObject](#) = null [get, private set]

Private Member Functions

- void [Awake \(\)](#)
- void [ToggleGazeRaycastingVrStatusChange \(VRManager.VRStatus vrStatus\)](#)
Callback for VRManagers XRStatusUpdate. Turns raycast on or off based on if VR is enabled.
- void [GazeRaycast \(\)](#)
Simple raycast based on gaze direction. Only interacts with first object hit. Can't handle overlapping objects yet.
- void [OnNothingHit \(\)](#)
Handles what happens if no object got hit by raycast but previously and object was being hit. -> Calls OnRaycastExit.
- void [OnObjectHit \(GameObject new_hit\)](#)
Handles what happens when an object is hit by raycast.

4.24.1 Detailed Description

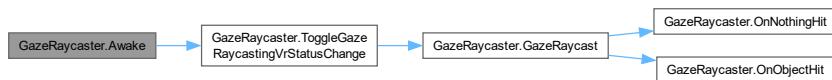
Creates a raycast from current eyedata gaze direction. If an object with the [RaycastHitHandler](#) component is hit by raycast this script calls the [OnRaycastEnter](#) and [OnRaycastExit](#) methods on entry and exit of the object. This allows objects to react to being looked at by the participant.

4.24.2 Member Function Documentation

4.24.2.1 Awake()

```
void GazeRaycaster.Awake ( ) [private]
```

Here is the call graph for this function:

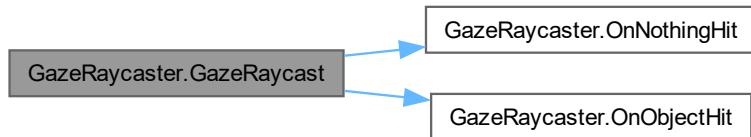


4.24.2.2 GazeRaycast()

```
void GazeRaycaster.GazeRaycast ( ) [private]
```

Simple raycast based on gaze direction. Only interacts with first object hit. Can't handle overlapping objects yet.

Here is the call graph for this function:



Here is the caller graph for this function:



4.24.2.3 OnNothingHit()

```
void GazeRaycaster.OnNothingHit ( ) [private]
```

Handles what happens if no object got hit by raycast but previously an object was being hit. -> Calls OnRaycastExit.

Here is the caller graph for this function:



4.24.2.4 OnObjectHit()

```
void GazeRaycaster.OnObjectHit (
    GameObject new_hit ) [private]
```

Handles what happens when an object is hit by raycast.

Parameters

<code>new_hit</code>	<input type="text"/>
----------------------	----------------------

Here is the caller graph for this function:

**4.24.2.5 ToggleGazeRaycastingVrStatusChange()**

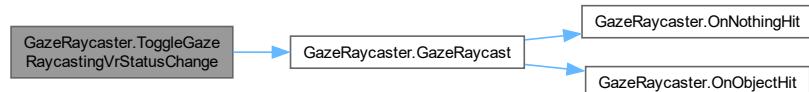
```
void GazeRaycaster.ToggleGazeRaycastingVrStatusChange (
    VRManager::VRStatus vrStatus ) [private]
```

Callback for VRManagers XRStatusUpdate. Turns raycast on or off based on if VR is enabled.

Parameters

<code>vrStatus</code>	New status of VR.
-----------------------	-------------------

Here is the call graph for this function:



Here is the caller graph for this function:

**4.24.3 Member Data Documentation****4.24.3.1 Instance**

`GazeRaycaster` `GazeRaycaster.Instance` [static]

4.24.4 Property Documentation

4.24.4.1 CurrentlyHitObject

```
GameObject GazeRaycaster.CurrentlyHitObject = null [get], [private set]
```

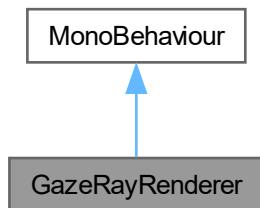
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/[GazeRaycaster.cs](#)

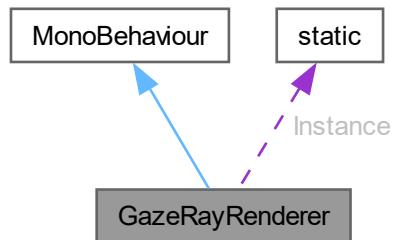
4.25 GazeRayRenderer Class Reference

Derived from SRanipal Example Scripts. This script displays the current gaze of the participant as a colored ray within the experimentor-view of the application.

Inheritance diagram for GazeRayRenderer:



Collaboration diagram for GazeRayRenderer:



Public Member Functions

- void [RenderCurrentGazeAsRay \(\)](#)

Renders current Gaze as ray. Turns ray blue if gaze data is valid, gray of gaze data is invalid.

Public Attributes

- int `_LengthOfRay` = 1000

Static Public Attributes

- static `GazeRayRenderer Instance`

Private Member Functions

- void `Awake ()`
- void `ToggleGazeRenderingOnVrStatusChange (VRManager.VRStatus vrStatus)`
Callback for VRManagers XRStatusUpdate Event. Enables GazeRay when VR is enabled. Disables Gazeray when VR is disabled.

Private Attributes

- LineRenderer `_GazeRayRenderer`
- Color `_ray_color` = Color.red
- GameObject `_gazeSphere`
- Renderer `_gazeSphereRenderer`

4.25.1 Detailed Description

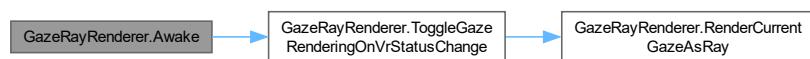
Derived from SRanipal Example Scripts. This script displays the current gaze of the participant as a colored ray within the experimentor-view of the application.

4.25.2 Member Function Documentation

4.25.2.1 Awake()

```
void GazeRayRenderer.Awake ( ) [private]
```

Here is the call graph for this function:



4.25.2.2 RenderCurrentGazeAsRay()

```
void GazeRayRenderer.RenderCurrentGazeAsRay ( )
```

Renders current Gaze as ray. Turns ray blue if gaze data is valid, gray of gaze data is invalid.

Here is the caller graph for this function:



4.25.2.3 ToggleGazeRenderingOnVrStatusChange()

```
void GazeRayRenderer.ToggleGazeRenderingOnVrStatusChange ( VRManager::VRStatus vrStatus ) [private]
```

Callback for VRManagers XRStatusUpdate Event. Enables GazeRay when VR is enabled. Disables Gazeray when VR is disabled.

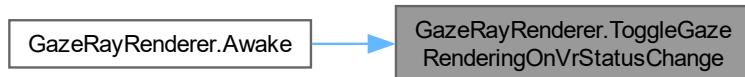
Parameters

vrStatus	<input type="button" value=""/>
----------	---------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



4.25.3 Member Data Documentation

4.25.3.1 `_GazeRayRenderer`

```
LineRenderer GazeRayRenderer._GazeRayRenderer [private]
```

4.25.3.2 `_gazeSphere`

```
GameObject GazeRayRenderer._gazeSphere [private]
```

4.25.3.3 `_gazeSphereRenderer`

```
Renderer GazeRayRenderer._gazeSphereRenderer [private]
```

4.25.3.4 `_LengthOfRay`

```
int GazeRayRenderer._LengthOfRay = 1000
```

4.25.3.5 `_ray_color`

```
Color GazeRayRenderer._ray_color = Color.red [private]
```

4.25.3.6 `Instance`

```
GazeRayRenderer GazeRayRenderer.Instance [static]
```

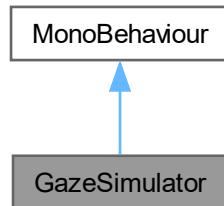
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/[GazeRayRenderer.cs](#)

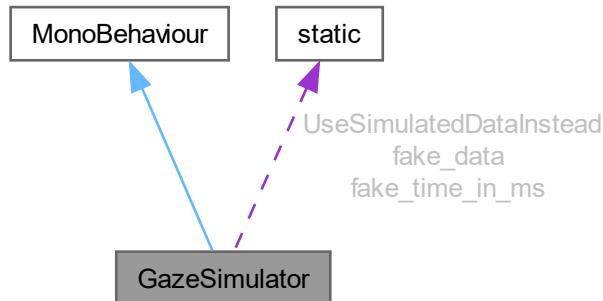
4.26 GazeSimulator Class Reference

Another test-script. The code generates fixations at a specified Vector3. Instead of creating a eventdetection object here the fake_data is used in the eyerecorder class when the UseSimulatedDataInstead flag is set to true.

Inheritance diagram for GazeSimulator:



Collaboration diagram for GazeSimulator:



Static Public Member Functions

- static [EyeTrackingDataSample GetNextFakeSample \(\)](#)

Used by Eyerecorder to get next generated datasample during recording when UseSimulatedDataInstead flag is set to true.

Public Attributes

- Vector3 [TestRayDirection](#)

Static Public Attributes

- static bool `UseSimulatedDataInstead` = false

Private Member Functions

- void `Start` ()
- void `OnDrawGizmos` ()

debug code used to draw a vector in the scene view based on TestRayDirection. Allows to visually check what vector where a given vector is pointing to.
- void `GenerateFixation` (Vector3 center, int duration=150, int validityPercentge=100, float dispersionLimit=-1, float velocityLimit=-1)

Generates a fixation (or dwell) based on the provided parameters.

Static Private Attributes

- static List< `EyeTrackingDataSample` > `fake_data`
- static int `fake_time_in_ms` = 8

4.26.1 Detailed Description

Another test-script. The code generates fixations at a specified Vector3. Instead of creating a eventdetection object here the `fake_data` is used in the eyerecorder class when the `UseSimulatedDataInstead` flag is set to true.

4.26.2 Member Function Documentation

4.26.2.1 GenerateFixation()

```
void GazeSimulator.GenerateFixation (
    Vector3 center,
    int duration = 150,
    int validityPercentge = 100,
    float dispersionLimit = -1,
    float velocityLimit = -1 ) [private]
```

Generates a fixation (or dwell) based on the provided parameters.

Parameters

<code>center</code>	Center point of generated datasamples
<code>duration</code>	Duration of Fixation/Dwell
<code>validityPercentge</code>	Percentage of valid datasamples.
<code>dispersionLimit</code>	Dispersion limit for generated datasamples
<code>velocityLimit</code>	Velocity limit for generated datasamples

Here is the caller graph for this function:



4.26.2.2 GetNextFakeSample()

```
static EyeTrackingDataSample GazeSimulator.GetNextFakeSample ( ) [static]
```

Used by Eyerecorder to get next generated datasample during recording when UseSimulatedDataInstead flag is set to true.

Returns

Returns next generated DataSample.

4.26.2.3 OnDrawGizmos()

```
void GazeSimulator.OnDrawGizmos ( ) [private]
```

debug code used to draw a vector in the scene view based on TestRayDirection. Allows to visually check what vector where a given vector is pointing to.

4.26.2.4 Start()

```
void GazeSimulator.Start ( ) [private]
```

Here is the call graph for this function:



4.26.3 Member Data Documentation

4.26.3.1 fake_data

```
List<EyeTrackingDataSample> GazeSimulator.fake_data [static], [private]
```

4.26.3.2 fake_time_in_ms

```
int GazeSimulator.fake_time_in_ms = 8 [static], [private]
```

4.26.3.3 TestRayDirection

```
Vector3 GazeSimulator.TestRayDirection
```

4.26.3.4 UseSimulatedDataInstead

```
bool GazeSimulator.UseSimulatedDataInstead = false [static]
```

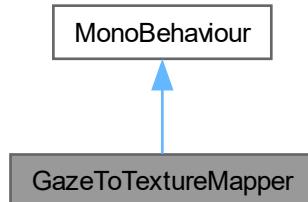
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/Simulation/GazeSimulator.cs

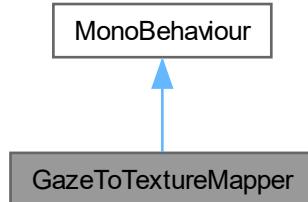
4.27 GazeToTextureMapper Class Reference

Proof of concept class. Derived from shader code of Unity PanoramicShader. Allows to map the gaze direction onto pixel coordinates of the skybox-texture. Can be used in the future to create different types of visualizations.

Inheritance diagram for GazeToTextureMapper:



Collaboration diagram for GazeToTextureMapper:



Public Attributes

- Vector3 `direction` = new Vector3(25, 0, 0)
- int `VideoWidth` = 4096
- int `VideoHeight` = 2048
- int `texelX`
- int `texelY`

Private Member Functions

- void `OnDrawGizmos()`
- Vector2 `VectorToUV` (Vector3 coords)

4.27.1 Detailed Description

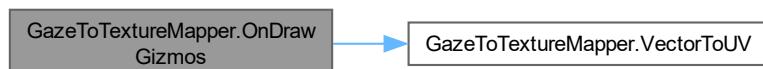
Proof of concept class. Derived from shader code of Unity PanoramicShader. Allows to map the gaze direction onto pixel coordinates of the skybox-texture. Can be used in the future to create different types of visualizations.

4.27.2 Member Function Documentation

4.27.2.1 OnDrawGizmos()

```
void GazeToTextureMapper.OnDrawGizmos ( ) [private]
```

Here is the call graph for this function:



4.27.2.2 VectorToUV()

```
Vector2 GazeToTextureMapper.VectorToUV (
    Vector3 coords) [private]
```

Here is the caller graph for this function:



4.27.3 Member Data Documentation

4.27.3.1 direction

```
Vector3 GazeToTextureMapper.direction = new Vector3(25, 0, 0)
```

4.27.3.2 texelX

```
int GazeToTextureMapper.texelX
```

4.27.3.3 texelY

```
int GazeToTextureMapper.texelY
```

4.27.3.4 VideoHeight

```
int GazeToTextureMapper.VideoHeight = 2048
```

4.27.3.5 VideoWidth

```
int GazeToTextureMapper.VideoWidth = 4096
```

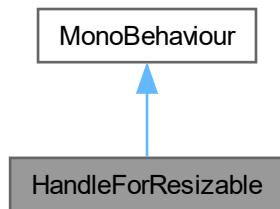
The documentation for this class was generated from the following file:

- Assets/Scripts/Logging and Visualization/[GazeToTextureMapper.cs](#)

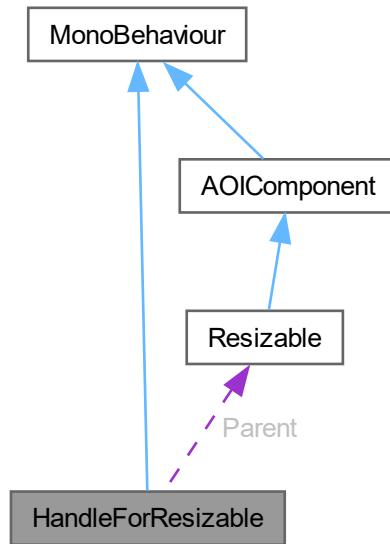
4.28 HandleForResizable Class Reference

This script is attached to every AOI-Handle. It signals the parent object that a handle has been clicked and resizing should start. Requires the parent to have the "Resizable" Component.

Inheritance diagram for HandleForResizable:



Collaboration diagram for HandleForResizable:



Public Member Functions

- void [OnMouseDown \(\)](#)
- void [OnMouseUp \(\)](#)

Public Attributes

- [Resizable Parent](#)

Private Member Functions

- void [Start \(\)](#)

4.28.1 Detailed Description

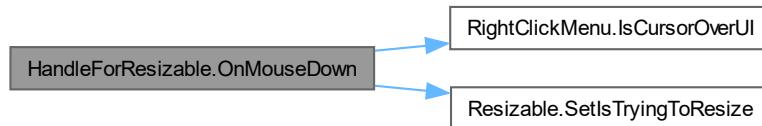
This script is attached to every AOI-Handle. It signals the parent object that a handle has been clicked and resizing should start. Requires the parent to have the "Resizable" Component.

4.28.2 Member Function Documentation

4.28.2.1 OnMouseDown()

```
void HandleForResizable.OnMouseDown ( )
```

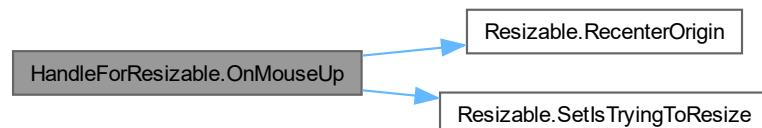
Here is the call graph for this function:



4.28.2.2 OnMouseUp()

```
void HandleForResizable.OnMouseUp ( )
```

Here is the call graph for this function:



4.28.2.3 Start()

```
void HandleForResizable.Start ( ) [private]
```

4.28.3 Member Data Documentation

4.28.3.1 Parent

`Resizable` `HandleForResizable.Parent`

The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/AOI Handles/[HandleForResizable.cs](#)

4.29 Helper Class Reference

Static Public Member Functions

- static T [MostCommon< T >](#) (this IEnumerable< T > list)

Used to determine which AOI a fixation should be assigned to.

4.29.1 Member Function Documentation

4.29.1.1 [MostCommon< T >\(\)](#)

```
static T Helper.MostCommon< T > (
    this IEnumerable< T > list ) [static]
```

Used to determine which AOI a fixation should be assigned to.

Template Parameters

T	
---	--

Parameters

list	
------	--

Returns

Returns the value that has the most occurrences within a list

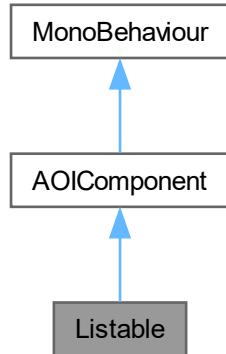
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/[EventDetection.cs](#)

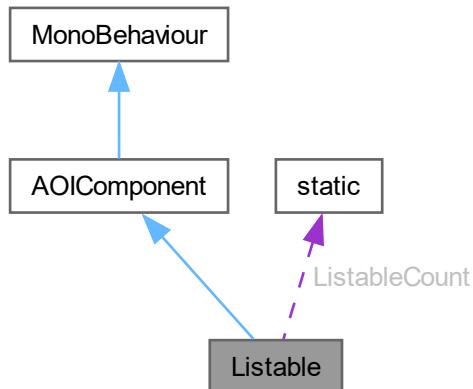
4.30 Listable Class Reference

This script manages the name of the AOI and its representation in the AOIList in the UI.

Inheritance diagram for Listable:



Collaboration diagram for Listable:



Public Member Functions

- void [SetName](#) (string newname)
Changes name of AOI and updates text in list.

Static Public Member Functions

- static implicit [operator string](#) ([Listable](#) listable)

Public Attributes

- GameObject [ItemPrefab](#)

Static Public Attributes

- static int [ListableCount](#) = 0

Protected Member Functions

- override void [Start](#) ()
- override void [OnDestroy](#) ()
- override void [OnActivate](#) ()
changes color of item in AOI list when AOI is being activated.
- override void [OnDeactivate](#) ()
changes color of item in AOI list when AOI is being de-activated.

Protected Member Functions inherited from [AOIComponent](#)

- virtual void [Start](#) ()
- virtual void [ActiveAOIchanged](#) (GameObject newAoi)
Callback for AOIManagers NewActiveAOI-Event. Called whenever user clicks on new AOI.
- abstract void [OnActivate](#) ()
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.
- abstract void [OnDeactivate](#) ()
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.
- virtual void [OnDestroy](#) ()

Properties

- string [Name](#) = "" [get, private set]

Private Member Functions

- void [UpdateNameFromInputField](#) (string newname)
Callback for UI-Inputfield. Gets called when edit is being completed. Changes name of AOI to whatever user has specified in inputfield.
- void [OnListItemClick](#) (BaseEventData eventData)
Callback for click event on AOI-list item. Implements single-click and double-click behavior.

Private Attributes

- GameObject [_itemInstance](#)
- Transform [_contentContainer](#)
- TMPro.TMP_InputField [_inputField](#)

4.30.1 Detailed Description

This script manages the name of the AOI and its representation in the AOIList in the UI.

4.30.2 Member Function Documentation

4.30.2.1 OnActivate()

```
override void Listable.OnActivate ( ) [protected], [virtual]
```

changes color of item in AOI list when AOI is being activated.

Implements [AOIComponent](#).

4.30.2.2 OnDeactivate()

```
override void Listable.OnDeactivate ( ) [protected], [virtual]
```

changes color of item in AOI list when AOI is being de-activated.

Implements [AOIComponent](#).

4.30.2.3 OnDestroy()

```
override void Listable.OnDestroy ( ) [protected], [virtual]
```

Reimplemented from [AOIComponent](#).

4.30.2.4 OnListItemClick()

```
void Listable.OnListItemClick (
    BaseEventData eventData) [private]
```

Callback for click event on AOI-list item. Implements single-click and double-click behavior.

Parameters

eventData	
-----------	--

Here is the call graph for this function:



Here is the caller graph for this function:



4.30.2.5 operator string()

```
static implicit Listable.operator string (
    Listable listable ) [static]
```

4.30.2.6 SetName()

```
void Listable.SetName (
    string newname )
```

Changes name of AOI and updates text in list.

Parameters

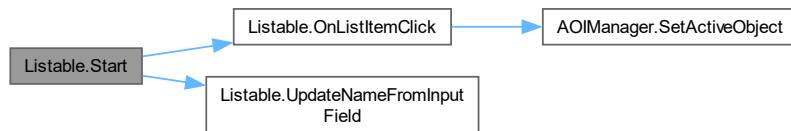
<i>newname</i>	New name of AOI
----------------	-----------------

4.30.2.7 Start()

```
override void Listable.Start ( ) [protected], [virtual]
```

Reimplemented from [AOIComponent](#).

Here is the call graph for this function:



4.30.2.8 `UpdateNameFromInputField()`

```
void Listable.UpdateNameFromInputField (
    string newname ) [private]
```

Callback for UI-Inputfield. Gets called when edit is being completed. Changes name of AOI to whatever user has specified in inputfield.

Parameters

<code>newname</code>	<input type="text"/>
----------------------	----------------------

Here is the caller graph for this function:



4.30.3 Member Data Documentation

4.30.3.1 `_contentContainer`

```
Transform Listable._contentContainer [private]
```

4.30.3.2 `_inputField`

```
TMPro.TMP_InputField Listable._inputField [private]
```

4.30.3.3 _itemInstance

```
GameObject Listable._itemInstance [private]
```

4.30.3.4 ItemPrefab

```
GameObject Listable.ItemPrefab
```

4.30.3.5 ListableCount

```
int Listable.ListableCount = 0 [static]
```

4.30.4 Property Documentation

4.30.4.1 Name

```
string Listable.Name = "" [get], [private set]
```

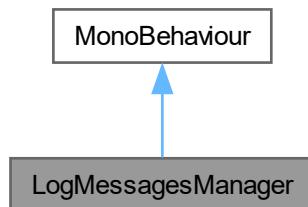
The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/AOI Components/[Listable.cs](#)

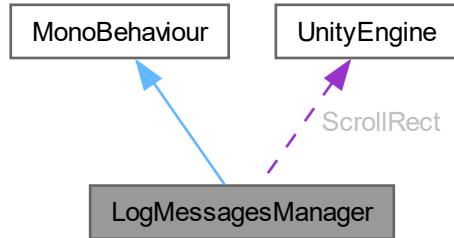
4.31 LogMessagesManager Class Reference

Script managing the creation of logs in the error/debug log console. Console can be opened and closed by pressing 'F1'.

Inheritance diagram for LogMessagesManager:



Collaboration diagram for LogMessagesManager:



Public Member Functions

- void [OpenLogPanel \(\)](#)
- void [CloseLogPanel \(\)](#)

Public Attributes

- GameObject [LogPanel](#)
- GameObject [LogEntryPrefab](#)
- UnityEngine.UI.ScrollRect [ScrollView](#)
- Transform [ContentContainer](#)

Properties

- static [LogMessagesManager Instance](#) [get, private set]

Private Member Functions

- void [Awake \(\)](#)
- void [Update \(\)](#)
- void [OnLogMessageReceived \(string logString, string stackTrace, LogType type\)](#)

Callback for unity's Application.logMessageReceived event. Creates log entry from prefab and displays it in log console.

4.31.1 Detailed Description

Script managing the creation of logs in the error/debug log console. Console can be opened and closed by pressing 'F1'.

4.31.2 Member Function Documentation

4.31.2.1 Awake()

```
void LogMessagesManager.Awake ( ) [private]
```

Here is the call graph for this function:



4.31.2.2 CloseLogPanel()

```
void LogMessagesManager.CloseLogPanel ( )
```

4.31.2.3 OnLogMessageReceived()

```
void LogMessagesManager.OnLogMessageReceived (
    string logString,
    string stackTrace,
    LogType type ) [private]
```

Callback for unity's Application.logMessageReceived event. Creates log entry from prefab and displays it in log console.

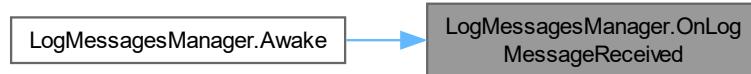
Parameters

<i>logString</i>	String of the log-message.
<i>stackTrace</i>	
<i>type</i>	Type of log-message, whether its e.g. an error, a warning, just a debug log etc...

Here is the call graph for this function:



Here is the caller graph for this function:



4.31.2.4 OpenLogPanel()

```
void LogMessagesManager.OpenLogPanel( )
```

Here is the caller graph for this function:



4.31.2.5 Update()

```
void LogMessagesManager.Update( ) [private]
```

4.31.3 Member Data Documentation

4.31.3.1 ContentContainer

```
Transform LogMessagesManager.ContentContainer
```

4.31.3.2 LogEntryPrefab

```
GameObject LogMessagesManager.LogEntryPrefab
```

4.31.3.3 LogPanel

```
GameObject LogMessagesManager.LogPanel
```

4.31.3.4 ScrollRect

```
UnityEngine.UI.ScrollRect LogMessagesManager.ScrollRect
```

4.31.4 Property Documentation

4.31.4.1 Instance

```
LogMessagesManager LogMessagesManager.Instance [static], [get], [private set]
```

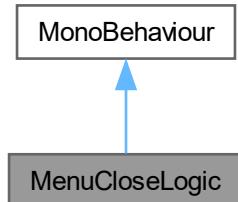
The documentation for this class was generated from the following file:

- Assets/Scripts/Application/[LogMessagesManager.cs](#)

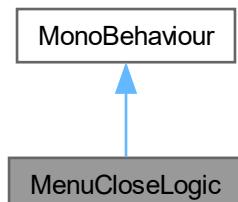
4.32 MenuCloseLogic Class Reference

Tiny little script thats attached to everything that should be "closed" when the user clicks outside of it::s bounds.
Mostly used for menus that should dissapear when user clicks outside of them.

Inheritance diagram for MenuCloseLogic:



Collaboration diagram for MenuCloseLogic:



Private Member Functions

- void [Update \(\)](#)

4.32.1 Detailed Description

Tiny little script that's attached to everything that should be "closed" when the user clicks outside of it's bounds. Mostly used for menus that should disappear when user clicks outside of them.

4.32.2 Member Function Documentation

4.32.2.1 Update()

```
void MenuCloseLogic.Update ( ) [private]
```

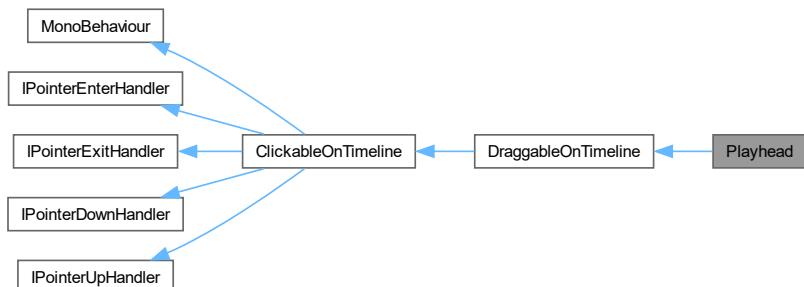
The documentation for this class was generated from the following file:

- Assets/Scripts/Application/[MenuCloseLogic.cs](#)

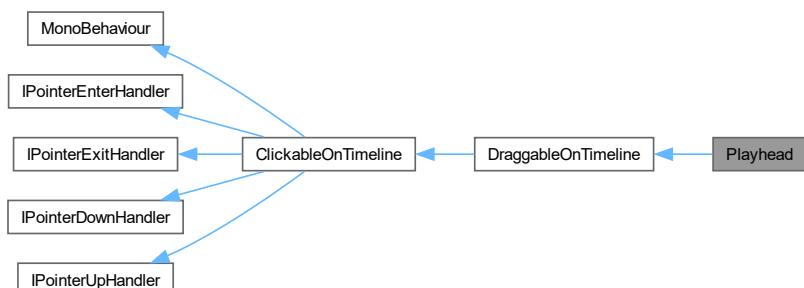
4.33 Playhead Class Reference

Implements the playhead. Inherits [DraggableOnTimeline](#).

Inheritance diagram for Playhead:



Collaboration diagram for Playhead:



Public Member Functions

- override void [SetAnchoredX](#) (float new_x)
Allows the timeline to change to position of the playhead during video playback.

Public Member Functions inherited from [DraggableOnTimeline](#)

- delegate void [MovementCompleted](#) ()
- override void [OnPointerDown](#) (PointerEventData eventData)
If an item on the timeline has been clicked it marks itself as clicked and takes the hover away from the timeline.
- virtual void [SetAnchoredX](#) (float new_x)

Public Member Functions inherited from [ClickableOnTimeline](#)

- virtual void [OnPointerEnter](#) (PointerEventData eventData)
When the cursor hover above an item on the timeline the item marks itself as hovered.
- virtual void [OnPointerExit](#) (PointerEventData eventData)
If the cursor leaves the area of the clickable item but has previously been clicked and the mouse is still pressed down, the item makes sure that the timeline still doesn't register as being hovered, because the user is still interacting with the item itself, since the mouse is still pressed down (the user might e.g. be trying to drag an item across the timeline.). If the cursor leaves the area of the item and the item has not been clicked, it sets itself as not hovered and hands over the hover to the timeline.
- virtual void [OnPointerDown](#) (PointerEventData eventData)
If an item on the timeline has been clicked it marks itself as clicked and takes the hover away from the timeline.
- virtual void [OnPointerUp](#) (PointerEventData eventData)
On Pointer up the item registers itself as not clicked anymore, allows edge snapping of the timeline abd handsover hover to the timeline.

Public Attributes

- RectTransform [CurrentFrameLabelWithBG](#)
- TMPro.TextMeshProUGUI [CurrentFrameLabel](#)

Public Attributes inherited from [DraggableOnTimeline](#)

- [MovementCompleted](#) [MovementCompletedSubscribers](#)

Protected Member Functions

- override void [Update](#) ()
After a clickable item on the timeline has been clicked, for as long as the mouse is still down, the update function makes sure that the timeline never registers as being "hovered", since the user is hovering an item on the timeline, not the timeline itself.
- override void [OnPositionChange](#) ()
Called when user changes position of playhead by dragging it along the timeline. Sets the targetframe of the timeline based on the new position of the playhead.
- override void [OnMovementCompleted](#) ()
Called when user releases mouseclick after having moved the playhead. Tells timeline to update targetframe based on playhead position as well as updating the video based on new target frame.
- override bool [MovementCondition](#) ()

Protected Member Functions inherited from DraggableOnTimeline

- override void [Start \(\)](#)
- override void [Update \(\)](#)

After a clickable item on the timeline has been clicked, for as long as the mouse is still down, the update function makes sure that the timeline never registeres as being "hovered", since the user is hovering and item on the timeline, not the timeline itself.

- virtual void [OnPositionChange \(\)](#)
- virtual void [OnMovementCompleted \(\)](#)
- virtual bool [MovementCondition \(\)](#)

- virtual void [Start \(\)](#)
- virtual void [Update \(\)](#)

After a clickable item on the timeline has been clicked, for as long as the mouse is still down, the update function makes sure that the timeline never registeres as being "hovered", since the user is hovering and item on the timeline, not the timeline itself.

Additional Inherited Members

Protected Attributes inherited from DraggableOnTimeline

- Vector2 [_mouseOffset](#)

Protected Attributes inherited from ClickableOnTimeline

- RectTransform [_rect](#)

Properties inherited from DraggableOnTimeline

- Canvas [Canvas](#) [get, protected set]
- bool [IsBeingDragged](#) = false [get, protected set]

Properties inherited from ClickableOnTimeline

- Timeline [Timeline](#) [get, protected set]
- bool [IsHovered](#) = false [get, protected set]
- bool [IsClickedAndMouseStillDown](#) = false [get, protected set]

4.33.1 Detailed Description

Implements the playhead. Inherits DraggableOnTimeline.

4.33.2 Member Function Documentation

4.33.2.1 MovementCondition()

```
override bool Playhead.MovementCondition ( ) [protected], [virtual]
```

Reimplemented from [DraggableOnTimeline](#).

4.33.2.2 OnMovementCompleted()

```
override void Playhead.OnMovementCompleted ( ) [protected], [virtual]
```

called when user releases mouseclick after having moved the playhead. Tells timeline to update targetframe based on playhead position as well as updating the video based on new target frame.

Reimplemented from [DraggableOnTimeline](#).

Here is the call graph for this function:



4.33.2.3 OnPositionChange()

```
override void Playhead.OnPositionChange ( ) [protected], [virtual]
```

Called when user changes position of playhead by dragging it along the timeline. Sets the targetframe of the timeline based on the new position of the playhead.

Reimplemented from [DraggableOnTimeline](#).

Here is the call graph for this function:



4.33.2.4 SetAnchoredX()

```
override void Playhead.SetAnchoredX ( float new_x ) [virtual]
```

Allows the timeline to change to position of the playhead during video playback.

Parameters

<i>new</i> ↪	
<i>_x</i>	

Reimplemented from [DraggableOnTimeline](#).

Here is the caller graph for this function:



4.33.2.5 Update()

```
override void Playhead.Update ( ) [protected], [virtual]
```

After a clickable item on the timeline has been clicked, for as long as the mouse is still down, the update function makes sure that the timeline never registeres as being "hovered", since the user is hovering and item on the timeline, not the timeline itself.

Reimplemented from [DraggableOnTimeline](#).

4.33.3 Member Data Documentation

4.33.3.1 CurrentFrameLabel

```
TMPRO.TextMeshProUGUI Playhead.CurrentFrameLabel
```

4.33.3.2 CurrentFrameLabelWithBG

```
RectTransform Playhead.CurrentFrameLabelWithBG
```

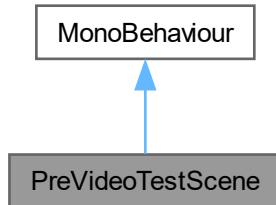
The documentation for this class was generated from the following file:

- Assets/Scripts/Video and Timeline/[Playhead.cs](#)

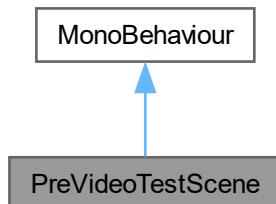
4.34 PreVideoTestScene Class Reference

Handles display of testscene after vr is enabled but before recording starts.

Inheritance diagram for PreVideoTestScene:



Collaboration diagram for PreVideoTestScene:



Private Member Functions

- void [Awake \(\)](#)
- void [OnPreVideoTestSceneToggled](#) (bool isEnabled)
Callback for VrSettingsMenus OnIsPreVideoTestSceneEnabledToggled-Event. Makes Spheres visible when Toggle is set to active and VR is enabled.
- void [OnVrStatusChange](#) (VRManager.VRStatus status)
Callback for VRManagers OnXRStatusUpdate-Event. Makes Spheres visible when VR is being enabled and user has option toggled.
- void [OnRecordingToggled](#) (bool is_recording)
Callback for Eyerecorders OnRecordingToggled Event. Hides Spheres when recording has been started.

4.34.1 Detailed Description

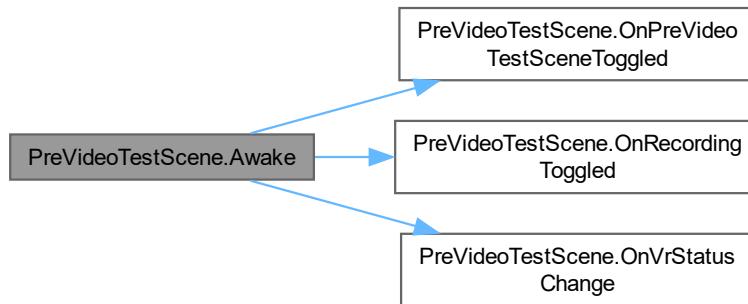
Handles display of testscene after vr is enabled but before recording starts.

4.34.2 Member Function Documentation

4.34.2.1 Awake()

```
void PreVideoTestScene.Awake ( ) [private]
```

Here is the call graph for this function:



4.34.2.2 OnPreVideoTestSceneToggled()

```
void PreVideoTestScene.OnPreVideoTestSceneToggled (
    bool isEnabled ) [private]
```

Callback for VrSettingsMenus OnIsPreVideoTestSceneEnabledToggled-Event. Makes Spheres visible when Toggle is set to active and VR is enabled.

Parameters

<code>isEnabled</code>	Bool whether or not user wants to user testscene.
------------------------	---

Here is the caller graph for this function:



4.34.2.3 OnRecordingToggled()

```
void PreVideoTestScene.OnRecordingToggled (
```

```
bool is_recording ) [private]
```

Callback for Eyerecorders OnRecordingToggled Event. Hides Spheres when recording has been started.

Parameters

<i>is_recording</i>	Bool whether recording has been started or turned off.
---------------------	--

Here is the caller graph for this function:



4.34.2.4 OnVrStatusChange()

```
void PreVideoTestScene.OnVrStatusChange ( VRManager::VRStatus status ) [private]
```

Callback for VRManagers OnXRStatusUpdate-Event. Makes Spheres visible when VR is being enabled and user has option toggled.

Parameters

<i>status</i>	New Status of VR Framework.
---------------	-----------------------------

Here is the caller graph for this function:



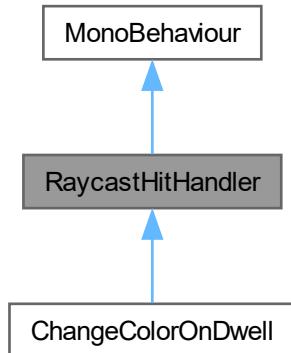
The documentation for this class was generated from the following file:

- Assets/Scripts/VR Management/[PreVideoTestScene.cs](#)

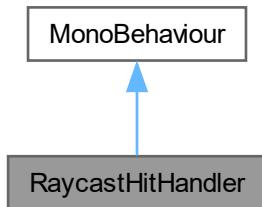
4.35 RaycastHitHandler Class Reference

Abstract class that defines OnRaycastEnter and OnRaycastExit for interaction with [GazeRaycaster](#). Should probably be an interface in the future.

Inheritance diagram for RaycastHitHandler:



Collaboration diagram for RaycastHitHandler:



Public Member Functions

- virtual void [OnRaycastEnter \(\)](#)
Called by GazeRaycaster when Dwell on object is started.
- virtual void [OnRaycastExit \(\)](#)
Called by GazeRaycaster when Dwell on object has ended.

Properties

- bool [IsCurrentlyHitByRaycast = false](#) [get, private set]

4.35.1 Detailed Description

Abstract class that defines OnRaycastEnter and OnRaycastExit for interaction with [GazeRaycaster](#). Should probably be an interface in the future.

4.35.2 Member Function Documentation

4.35.2.1 OnRaycastEnter()

```
virtual void RaycastHitHandler.OnRaycastEnter () [virtual]
```

Called by [GazeRaycaster](#) when Dwell on object is started.

Reimplemented in [ChangeColorOnDwell](#).

4.35.2.2 OnRaycastExit()

```
virtual void RaycastHitHandler.OnRaycastExit () [virtual]
```

Called by [GazeRaycaster](#) when Dwell on object has ended.

Reimplemented in [ChangeColorOnDwell](#).

4.35.3 Property Documentation

4.35.3.1 IsCurrentlyHitByRaycast

```
bool RaycastHitHandler.IsCurrentlyHitByRaycast = false [get], [private set]
```

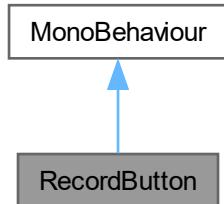
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/[RaycstHitHandler.cs](#)

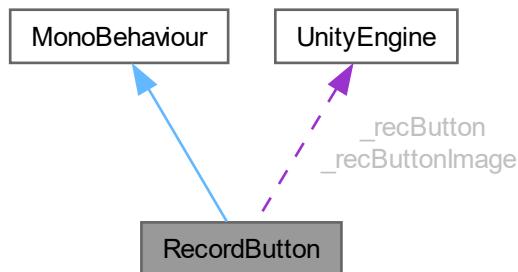
4.36 RecordButton Class Reference

Script attached to record button at the top right corner of UI. Handles sprite changes based on events.

Inheritance diagram for RecordButton:



Collaboration diagram for RecordButton:



Public Member Functions

- void [SetImageOnRecordingToggle](#) (bool state)

Callback for EyeRecorders RecordingToggled Event. Changes sprite based on whether recording is on or off.

Public Attributes

- Sprite [_offSprite](#)
- Sprite [_onSprite](#)

Private Member Functions

- void [Start](#) ()
- void [SetInteractabilityBasedOnVrStatus](#) (VRManager.VRStatus new_status)

Callback fro VRManagers XRStatusUpdate. Makes button interactable when VR is successfully enabled.

Private Attributes

- UnityEngine.UI.Button `_recButton`
- UnityEngine.UI.Image `_recButtonImage`

4.36.1 Detailed Description

Script attached to record button at the top right corner of UI. Handles sprite changes based on events.

4.36.2 Member Function Documentation

4.36.2.1 SetImageOnRecordingToggle()

```
void RecordButton.SetImageOnRecordingToggle (
    bool state )
```

Callback for EyeRecorders RecordingToggled Event. Changes sprite based on whether recording is on or off.

Parameters

<code>state</code>	<input type="text"/>
--------------------	----------------------

Here is the caller graph for this function:



4.36.2.2 SetInteractabilityBasedOnVrStatus()

```
void RecordButton.SetInteractabilityBasedOnVrStatus (
    VRManager::VRStatus new_status ) [private]
```

Callback fro VRManagers XRStatusUpdate. Makes button interactable when VR is successfully enabled.

Parameters

<code>new_status</code>	New status of VR.
-------------------------	-------------------

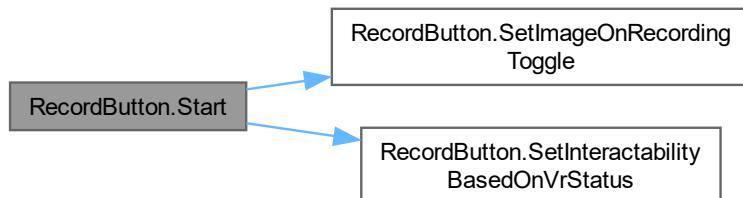
Here is the caller graph for this function:



4.36.2.3 Start()

```
void RecordButton.Start ( ) [private]
```

Here is the call graph for this function:



4.36.3 Member Data Documentation

4.36.3.1 _offSprite

```
Sprite RecordButton._offSprite
```

4.36.3.2 _onSprite

```
Sprite RecordButton._onSprite
```

4.36.3.3 _recButton

```
UnityEngine.UI.Button RecordButton._recButton [private]
```

4.36.3.4 _recButtonImage

```
UnityEngine.UI.Image RecordButton._recButtonImage [private]
```

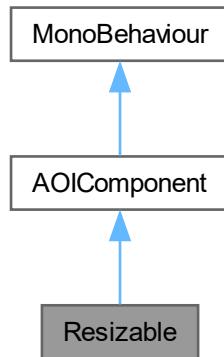
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/RecordButton.cs

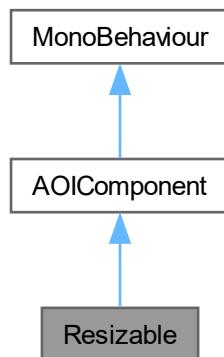
4.37 Resizable Class Reference

This script is attached to AOIs and is responsible for the creation of handles as well as resizing. Resizing happens when the user drags a handle of the currently selected AOI across the screen. Something of note: Resizing is done by changing the vertices of the AOIs mesh. Moving the vertices doesn't automatically recalculate the center (origin) of the game object, so this class manually recenters the objects origin during and after resizing is completed.

Inheritance diagram for Resizable:



Collaboration diagram for Resizable:



Public Member Functions

- void [SetIsTryingToResize](#) (HandleForResizable handle, bool value)
Called by the handle-gameobjects of the AOI. Allows handles to signal to AOI that resizing is happening and it should now start recalculating its mesh, or that resizing is now coming to an end.
- void [ShowHandles](#) ()
Show handles at corners of AOI.
- void [HideHandles](#) ()
Hide handles at corners of AOI.
- void [RecenterOrigin](#) ()
Recenters the origin of the object back to its mesh's center after changing vertices. This way a nice and consistent centerpoint is kept for rotating and positioning the AOIs around the Editor Camera.
- Vector3[] [RecenterOriginBasedOnKeyframeData](#) (Vector3[] newVertices)
Called by [Animation](#) Component when AOIs are loaded from a csv file.

Public Attributes

- GameObject [HandlePrefab](#)
- UnityEvent [ResizingCompleted](#)
- UnityEvent [ResizingStarted](#)

Protected Member Functions

- override void [Start](#) ()
- override void [OnActivate](#) ()
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.
- override void [OnDeactivate](#) ()
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.

Protected Member Functions inherited from [AOIComponent](#)

- virtual void [Start](#) ()
- virtual void [ActiveAOIchanged](#) (GameObject newAoi)
Callback for AOIManagers NewActiveAOI-Event. Called whenever user clicks on new AOI.
- abstract void [OnActivate](#) ()
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.
- abstract void [OnDeactivate](#) ()
Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.
- virtual void [OnDestroy](#) ()

Properties

- bool [IsTryingToResize](#) = false [get, protected set]
- Mesh [Mesh](#) [get, protected set]

Private Member Functions

- void [Update \(\)](#)
- void [OnDrawGizmos \(\)](#)
for debugging purposes
- void [Resize \(\)](#)

Resizes the AOIs mesh by moving each vertex to the position of its corresponding handle.

Private Attributes

- MeshCollider [_collider](#)
- Renderer [_renderer](#)
- Vector3[] [_vertices](#)
- List<[HandleForResizable](#)> [_handles](#) = new List<[HandleForResizable](#)>()

4.37.1 Detailed Description

This script is attached to AOIs and is responsible for the creation of handles as well as resizing. Resizing happens when the user drags a handle of the currently selected AOI across the screen. Something of note: Resizing is done by changing the vertices of the AOIs mesh. Moving the vertices doesn't automatically recalculate the center (origin) of the game object, so this class manually recenters the objects origin during and after resizing is completed.

4.37.2 Member Function Documentation

4.37.2.1 HideHandles()

```
void Resizable.HideHandles ( )
```

Hide handles at corners of AOI.

Here is the caller graph for this function:



4.37.2.2 OnActivate()

```
override void Resizable.OnActivate ( ) [protected], [virtual]
```

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user clicks on AOI.

Implements [AOIComponent](#).

Here is the call graph for this function:



4.37.2.3 OnDeactivate()

```
override void Resizable.OnDeactivate ( ) [protected], [virtual]
```

Method needs to be implemented by all AOI-Components. Specified how component reacts, when user de-selects AOI.

Implements [AOIComponent](#).

Here is the call graph for this function:



4.37.2.4 OnDrawGizmos()

```
void Resizable.OnDrawGizmos ( ) [private]
```

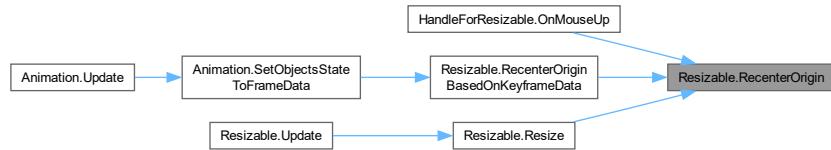
for debugging purposes

4.37.2.5 RecenterOrigin()

```
void Resizable.RecenterOrigin ( )
```

Recenters the origin of the object back to it's meshes center after changing vertices. This way a nice and consistent centerpoint is kept for rotating and positioning the AOIs around the Editor Camera.

Here is the caller graph for this function:



4.37.2.6 RecenterOriginBasedOnKeyframeData()

```
Vector3[] Resizable.RecenterOriginBasedOnKeyframeData (
    Vector3[] newVertices )
```

Called by [Animation](#) Component when AOIs are loaded from a csv file.

Parameters

<code>newVertices</code>	Vertices stored in keyframe data that need to be loaded into resizable component.
--------------------------	---

Returns

Here is the call graph for this function:



Here is the caller graph for this function:



4.37.2.7 Resize()

```
void Resizable.Resize ( ) [private]
```

Resizes the AOIs mesh by moving each vertex to the position of its corresponding handle.

Here is the call graph for this function:



Here is the caller graph for this function:



4.37.2.8 SetIsTryingToResize()

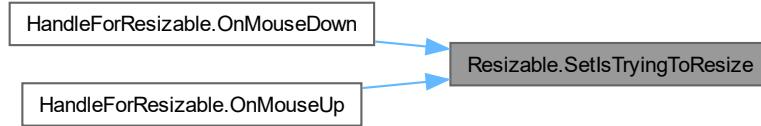
```
void Resizable.SetIsTryingToResize (
    HandleForResizable handle,
    bool value )
```

Called by the handle-gameobjects of the AOI. Allows handles to signal to AOI that resizing is happening and it should now start recalculating its mesh, or that resizing is now coming to an end.

Parameters

<i>handle</i>	The handle being dragged passes itself as parameter.
<i>value</i>	Bool to signal if movement of the handle is being started or stopped.

Here is the caller graph for this function:

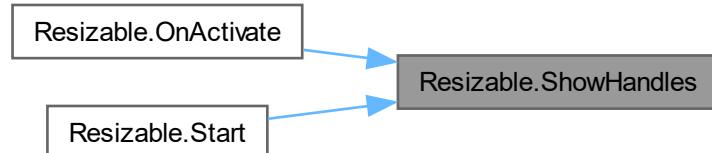


4.37.2.9 ShowHandles()

```
void Resizable.ShowHandles ( )
```

Show handles at corners of AOI.

Here is the caller graph for this function:



4.37.2.10 Start()

```
override void Resizable.Start ( ) [protected], [virtual]
```

Reimplemented from [AOIComponent](#).

Here is the call graph for this function:



4.37.2.11 Update()

```
void Resizable.Update ( ) [private]
```

Here is the call graph for this function:



4.37.3 Member Data Documentation

4.37.3.1 _collider

```
MeshCollider Resizable._collider [private]
```

4.37.3.2 _handles

```
List<HandleForResizable> Resizable._handles = new List<HandleForResizable>() [private]
```

4.37.3.3 _renderer

```
Renderer Resizable._renderer [private]
```

4.37.3.4 _vertices

```
Vector3 [] Resizable._vertices [private]
```

4.37.3.5 HandlePrefab

```
GameObject Resizable.HandlePrefab
```

4.37.3.6 ResizingCompleted

```
UnityEvent Resizable.ResizingCompleted
```

4.37.3.7 ResizingStarted

```
UnityEvent Resizable.ResizingStarted
```

4.37.4 Property Documentation

4.37.4.1 IsTryingToResize

```
bool Resizable.IsTryingToResize = false [get], [protected set]
```

4.37.4.2 Mesh

```
Mesh Resizable.Mesh [get], [protected set]
```

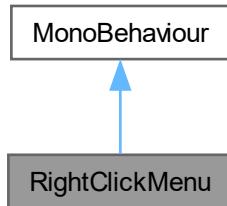
The documentation for this class was generated from the following file:

- Assets/Scripts/AOI Management/AOI Components/[Resizable.cs](#)

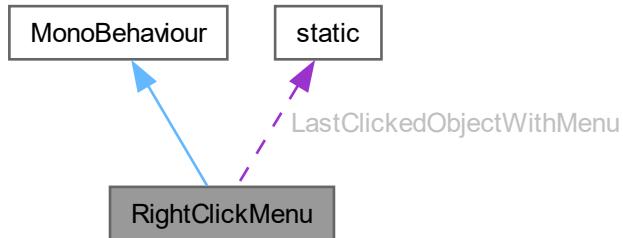
4.38 RightClickMenu Class Reference

This script can be attached to any gameobject to display a menu when the gameobject is right-clicked. Currently used on AOIs and keyframe-icons.

Inheritance diagram for RightClickMenu:



Collaboration diagram for RightClickMenu:



Public Member Functions

- void [AddItem](#) (string label, UnityEngine.Events.UnityAction action)
Adds entry to right click menu.
- void [RemoveItem](#) (string labelToDelete)
Removes an item from the right click menu based on the label.
- void [ShowMenu](#) ()
Shows right click menu close to cursor position.
- void [HideMenu](#) ()
Hides right click menu.

Static Public Member Functions

- static GameObject [GetChildGameObject](#) (GameObject fromGameObject, string withName)
helper function thats not necessary anymore. could be removed.
- static bool [IsCursorOverUI](#) ()
Helper function to check whether or not the cursor is currently over any UI element. For example, if the cursor is currently over the ui instead of the 360degree video a left click shouldnt spawn a new AOI etc. This helper is used throughout multiple scripts and should probably be moved to a different location.

Public Attributes

- Canvas [canvas](#)
- GameObject [_menuPrefab](#)
- GameObject [_itemPrefab](#)
- bool [AddDeleteOnStart](#)

Static Public Attributes

- static GameObject [LastClickedObjectWithMenu](#)

Protected Attributes

- Transform [_contentContainer](#)

Properties

- GameObject [Menu](#) [get, protected set]

Private Member Functions

- void [Awake](#) ()
- void [DeleteObjectUsingRightClickMenu](#) ()
Method to delete the object from its right click menu. deletes the menu-gameobject as well as the object itself.
- void [Update](#) ()
- bool [GotClicked](#) ()
Checks whether or not the object got clicked. Used together with Input.GetMouseButton(1) to determine click on object. Can't just use OnMouseDown because this needs to work with physics- as well as ui-objects.
- void [OnDestroy](#) ()

4.38.1 Detailed Description

This script can be attached to any gameobject to display a menu when the gameobject is right-clicked. Currently used on AOIs and keyframe-icons.

4.38.2 Member Function Documentation

4.38.2.1 AddItem()

```
void RightClickMenu.AddItem (
    string label,
    UnityEngine::Events::UnityAction action )
```

Adds entry to right click menu.

Parameters

<i>label</i>	Label for menu entry.
<i>action</i>	Callback for what happens when the entry is clicked.

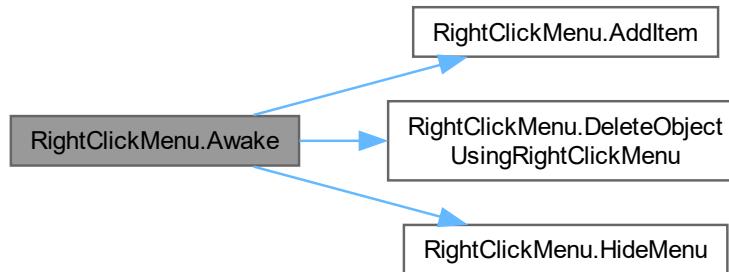
Here is the caller graph for this function:



4.38.2.2 Awake()

```
void RightClickMenu.Awake ( ) [private]
```

Here is the call graph for this function:

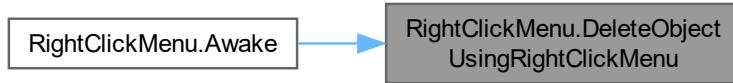


4.38.2.3 DeleteObjectUsingRightClickMenu()

```
void RightClickMenu.DeleteObjectUsingRightClickMenu ( ) [private]
```

Method to delete the object from its right click menu. deletes the menu-gameobject as well as the object itself.

Here is the caller graph for this function:



4.38.2.4 GetChildGameObject()

```
static GameObject RightClickMenu.GetChildGameObject (
    GameObject fromGameObject,
    string withName ) [static]
```

helper function thats not neccessary anymore. could be removed.

Parameters

<i>fromGameObject</i>	
<i>withName</i>	

Returns

4.38.2.5 GotClicked()

```
bool RightClickMenu.GotClicked ( ) [private]
```

Checks whether or not the object got clicked. Used together with Input.GetMouseButton(1) to determine click on object. Can't just used OnMouseDown because this needs to work with physics- as well as ui-objects.

Returns

Returns true if at time of method-call the cursor is above the object this script is attached to.

Here is the caller graph for this function:

**4.38.2.6 HideMenu()**

```
void RightClickMenu.HideMenu ( )
```

Hides right click menu.

Here is the caller graph for this function:

**4.38.2.7 IsCursorOverUI()**

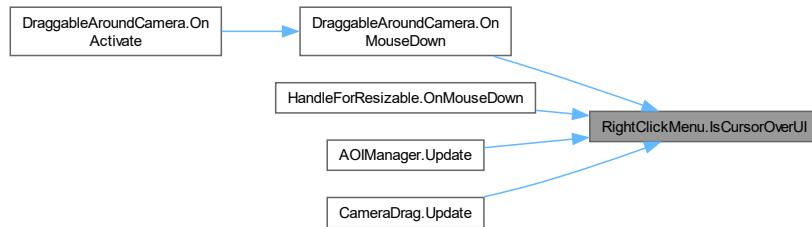
```
static bool RightClickMenu.IsCursorOverUI ( ) [static]
```

[Helper](#) function to check whether or not the cursor is currently over any UI element. For example, if the cursor is currently over the ui instead of the 360degree video a left click shouldnt spawn a new AOI etc. This helper is used throughout multiple scripts and should probably be moved to a different location.

Returns

True if cursor is currently over any UI element.

Here is the caller graph for this function:

**4.38.2.8 OnDestroy()**

```
void RightClickMenu.OnDestroy ( ) [private]
```

4.38.2.9 RemoveItem()

```
void RightClickMenu.RemoveItem (
    string labelToDelete )
```

Removes an item from the right click menu based on the label.

Parameters

<code>labelToDelete</code>	Entry to delete based on provided label.
----------------------------	--

4.38.2.10 ShowMenu()

```
void RightClickMenu.ShowMenu ( )
```

Shows right click menu close to cursor position.

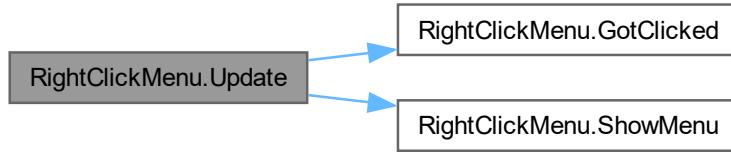
Here is the caller graph for this function:



4.38.2.11 Update()

```
void RightClickMenu.Update ( ) [private]
```

Here is the call graph for this function:



4.38.3 Member Data Documentation

4.38.3.1 _contentContainer

```
Transform RightClickMenu._contentContainer [protected]
```

4.38.3.2 _itemPrefab

```
GameObject RightClickMenu._itemPrefab
```

4.38.3.3 _menuPrefab

```
GameObject RightClickMenu._menuPrefab
```

4.38.3.4 AddDeleteOnStart

```
bool RightClickMenu.AddDeleteOnStart
```

4.38.3.5 canvas

```
Canvas RightClickMenu.canvas
```

4.38.3.6 LastClickedObjectWithMenu

```
GameObject RightClickMenu.LastClickedObjectWithMenu [static]
```

4.38.4 Property Documentation

4.38.4.1 Menu

```
GameObject RightClickMenu.Menu [get], [protected set]
```

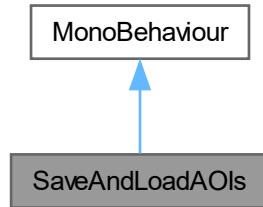
The documentation for this class was generated from the following file:

- Assets/Scripts/Application/[RightClickMenu.cs](#)

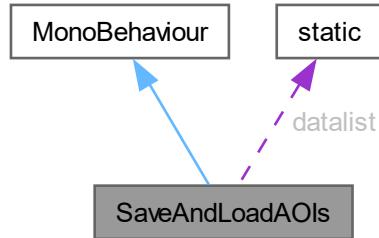
4.39 SaveAndLoadAOIs Class Reference

Handles saving and loading AOIs to and from a json file.

Inheritance diagram for SaveAndLoadAOIs:



Collaboration diagram for SaveAndLoadAOIs:



Classes

- class [AOISaveData](#)
- class [Datalist](#)

Public Member Functions

- void [Save \(\)](#)
Callback for Save Button OnButtonDown Event. Prompts user to select an existing file to overwrite or to declare new filename for AOI-save file. Converts all AOIs in current scene to json and stores information in selected file.
- void [OnLoadAOIButtonDown \(\)](#)
Callback for ButtonDown Event of Load AOIs Button. Prompts user to select a file from which to load AOIs.

Static Public Member Functions

- static void [LoadAois \(string path\)](#)
Loads AOIs from json file specified as parameter.

Static Public Attributes

- static [Datalist datalist = new\(\)](#)

4.39.1 Detailed Description

Handles saving and loading AOIs to and from a json file.

4.39.2 Member Function Documentation

4.39.2.1 LoadAois()

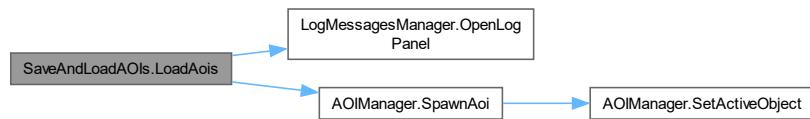
```
static void SaveAndLoadAOIs.LoadAois (
    string path ) [static]
```

Loads AOIs from json file specified as parameter.

Parameters

<i>path</i>	Path for json file to load AOIs from.
-------------	---------------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



4.39.2.2 OnLoadAOIButtonDown()

```
void SaveAndLoadAOIs.OnLoadAOIButtonDown ( )
```

Callback for ButtonDown Event of Load AOIs Button. Prompts user to select a file from which to load AOIs.

Here is the call graph for this function:



4.39.2.3 Save()

```
void SaveAndLoadAOIs.Save ( )
```

Callback for Save Button OnButtonDown Event. Prompts user to select an existing file to overwrite or to declare new filename for AOI-save file. Converts all AOIs in current scene to json and stores information in selected file.

4.39.3 Member Data Documentation

4.39.3.1 datalist

```
Datalist SaveAndLoadAOIs.datalist = new() [static]
```

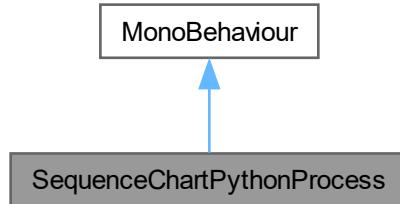
The documentation for this class was generated from the following file:

- Assets/Scripts/Project Management/[SaveAndLoadAOIs.cs](#)

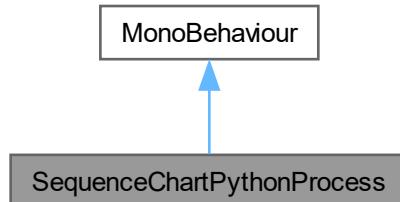
4.40 SequenceChartPythonProcess Class Reference

Calls python process that creates sequence chart from filteredData csv file.

Inheritance diagram for SequenceChartPythonProcess:



Collaboration diagram for SequenceChartPythonProcess:



Static Public Member Functions

- static void [CallSequenceChartScript](#) (string pathToDataCsv, string savePath)
Calls python process that creates sequence chart from filteredData csv file.

4.40.1 Detailed Description

Calls python process that creates sequence chart from filteredData csv file.

4.40.2 Member Function Documentation

4.40.2.1 CallSequenceChartScript()

```
static void SequenceChartPythonProcess.CallSequenceChartScript (
    string pathToDataCsv,
    string savePath ) [static]
```

Calls python process that creates sequence chart from filteredData csv file.

Parameters

<i>pathToDataCsv</i>	File location of filtered data csv file.
<i>savePath</i>	Folder where output graph should be stored.

Here is the caller graph for this function:



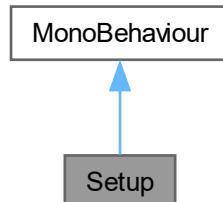
The documentation for this class was generated from the following file:

- Assets/Scripts/Logging and Visualization/SequenceChartPythonProcess.cs

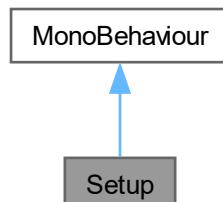
4.41 Setup Class Reference

Currently doesn't do too much. Sets target framerate of application to 120Hz.

Inheritance diagram for Setup:



Collaboration diagram for Setup:



Private Member Functions

- void [Start \(\)](#)
- void [SetFramerateOnVREnable \(VRManager.VRStatus status\)](#)

Callback fro VR Managers XRStatusUpdate Event. Attempts to set framerate even after vr is enabled. currently this is impossible, because as soon as vr is enabled the unity vr plugin takes over fps management and locks it to refresh rate of vr headsets screens. With htc vive pro eye that is 90Hz.

4.41.1 Detailed Description

Currently doesn't do too much. Sets target framerate of application to 120Hz.

4.41.2 Member Function Documentation

4.41.2.1 SetFramerateOnVREnable()

```
void Setup.SetFramerateOnVREnable (
    VRManager::VRStatus status ) [private]
```

Callback fro VR Managers XRStatusUpdate Event. Attempts to set framerate even after vr is enabled. currently this is impossible, because as soon as vr is enabled the unity vr plugin takes over fps management and locks it to refresh rate of vr headsets screens. With htc vive pro eye that is 90Hz.

Parameters

<i>status</i>	
---------------	--

Here is the caller graph for this function:



4.41.2.2 Start()

```
void Setup.Start ( ) [private]
```

Here is the call graph for this function:

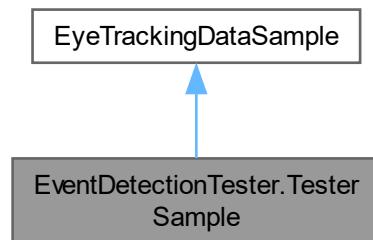


The documentation for this class was generated from the following file:

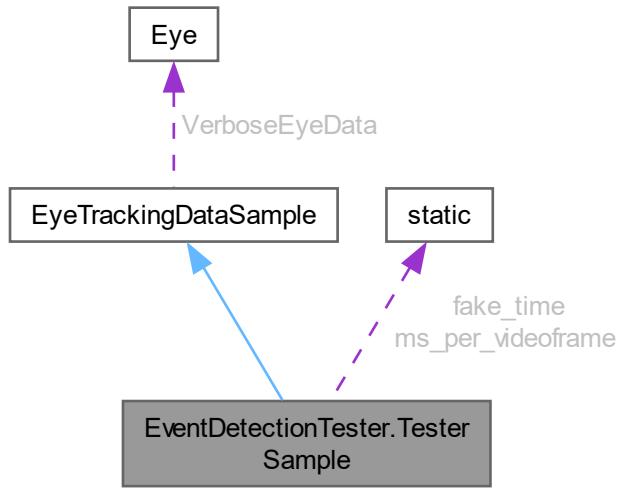
- Assets/Scripts/Application/[Setup.cs](#)

4.42 EventDetectionTester.TesterSample Class Reference

Inheritance diagram for EventDetectionTester.TesterSample:



Collaboration diagram for EventDetectionTester.TesterSample:



Public Types

- enum `aoi_test_selector` { `v1` , `v2` , `v3` }

Public Member Functions

- `TesterSample (aoi_test_selector v)`

Public Member Functions inherited from `EyeTrackingDataSample`

- `EyeTrackingDataSample (int timestamp, long frame, Vector3 direction, ViveSR.anipal.Eye.VerboseData verboseData, string aoiName="null")`
- `EyeTrackingDataSample ()`

Static Public Attributes

- static float `ms_per_videoframe` = 33.33333f
- static int `fake_time` = 8

Additional Inherited Members

Public Attributes inherited from `EyeTrackingDataSample`

- int `Timestamp`
- long `VideoFrame`
- Vector3 `GazeDirection`
- string `AoiName`
- ViveSR.anipal.Eye.VerboseData `VerboseEyeData`

4.42.1 Member Enumeration Documentation

4.42.1.1 aoi_test_selector

```
enum EventDetectionTester.TesterSample.aoi_test_selector
```

Enumerator

v1	
v2	
v3	

4.42.2 Constructor & Destructor Documentation

4.42.2.1 TesterSample()

```
EventDetectionTester.TesterSample.TesterSample (
    aoi_test_selector v )
```

4.42.3 Member Data Documentation

4.42.3.1 fake_time

```
int EventDetectionTester.TesterSample.fake_time = 8 [static]
```

4.42.3.2 ms_per_videoframe

```
float EventDetectionTester.TesterSample.ms_per_videoframe = 33.33333f [static]
```

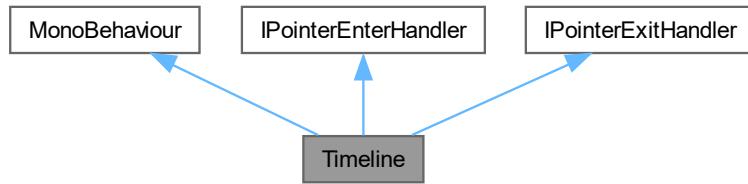
The documentation for this class was generated from the following file:

- Assets/Scripts/Eyetracking and Gaze Rendering/Simulation/[EventDetectionTester.cs](#)

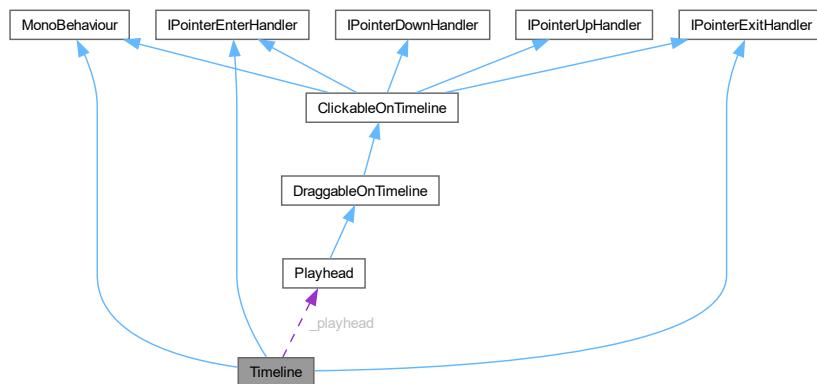
4.43 Timeline Class Reference

Implements functionality of timeline. Allows timeline to be scrubbed and to zoom in on the timeline to be able to move playhead with more precision to better place keyframes.

Inheritance diagram for Timeline:



Collaboration diagram for Timeline:



Public Member Functions

- void [FixPlayheadOvershoot \(\)](#)
Resets playheads position back to lefthand side of the timeline.
- void [ChangeTimelineScale \(float delta_zoom\)](#)
Handles “zooming” of timeline, adjusting start and endframe of timeline as well as playhead position based on new timeline scale.
- void [SyncTimelineWithVideo \(\)](#)
Can be used to sync timeline target frame and playhead position to current frame of video.
- void [OnPointerEnter \(PointerEventData eventData\)](#)
- void [OnPointerExit \(PointerEventData eventData\)](#)
Handles snapping, meaning when user exits timeline on left or right side of timeline the playhead should snap the start/end frame of timeline.
- void [SetBlockEdgeSnapping \(GameObject child, bool value\)](#)
Edge snapping might have to be disabled when children are currently handling input events that happen ontop of timeline.
- void [HoverHandoverWithChildren \(GameObject child, bool hover\)](#)
Allows children of timeline to signal if timeline is hovered or not.
- void [SetTargetFrame \(RectTransform childrect, bool updateplayhead, bool updatevideo\)](#)

- Used by children of timeline to set target frame based on childs position on timeline.
- void [SetTargetFrame](#) (long newframe, bool updateplayhead, bool updatevideo)
 - Used to update timeline to a new frame.
- void [SetVideoStartFrame](#) (float frame)
 - called by [VideoManager](#) when it has accurate information about what the first frame of the video is. Should maybe be handled by an event for better decoupling.

Public Attributes

- UnityEvent< long > [OnTargetFrameChange](#)
- RectTransform [PlayheadRect](#)
- TMPro.TextMeshProUGUI [StartFrameLabel](#)
- TMPro.TextMeshProUGUI [EndFrameLabel](#)

Properties

- static Timeline [Instance](#) [get, private set]
- float [Width](#) [get, private set]
- float [Height](#) [get, private set]
- float [ScrollScale](#) = 0.03f [get, private set]
- float [ZoomFactor](#) = 1 [get, private set]
- float [StartFrame](#) = 0 [get, private set]
- float [VideoStartFrame](#) = -1 [get, private set]
- float [EndFrame](#) [get, private set]
- float [WidthPerFrame](#) [get, private set]
- long [TargetFrame](#) = 0 [get, private set]
- bool [IsHovered](#) = false [get, private set]
- bool [BlockedEdgeSnapping](#) = false [get, private set]

Private Member Functions

- void [OnRectTransformDimensionsChange](#) ()
 - Callback for Eyerecorders RecordingToggled Event. Resets video and scrubber to 0 when recording starts.
- void [Awake](#) ()
 - Callback for VideoManagers VideoManagerInitialized Event. When videomanager is done loading the timeline can set the first and last frame.
- void [OnNewFrameInVideoPlayer](#) (long newframe)
 - Callback for VideoManagers OnNewFrame event. If theres a new frame displayed by the video and the timeline isnt currently being scrubbed by user, update playhead to new frame.
- void [OnVideoPlayerNewFrameCount](#) (ulong newframecount)
 - Callback for VideoManagers OnNewFrameCount event. The VideoPlayer might only know what exactly the last frame of the video is when playback approaches the end of the video. If the FrameCount changes, the timeline updates itself using SyncTimelineWithVideo and ChangeTimelineScale.
- void [Update](#) ()
 - Sets start frame of timeline and changes label at the left of timeline to display new start frame. Note: the StartFrame is not the first frame in the video, but the first frame of the video that is represented on the timeline.
- void [SetEndFrame](#) (float frame)
 - Sets end frame of timeline and changes label at the right of timeline to display new endframe. Note: the EndFrame is not the last frame in the video, but the last frame of the video that is represented on the timeline.

Private Attributes

- Playhead _playhead

4.43.1 Detailed Description

Implements functionality of timeline. Allows timeline to be scrubbed and to zoom in on the timeline to be able to move playhead with more precision to better place keyframes.

4.43.2 Member Function Documentation

4.43.2.1 Awake()

```
void Timeline.Awake ( ) [private]
```

4.43.2.2 ChangeTimelineScale()

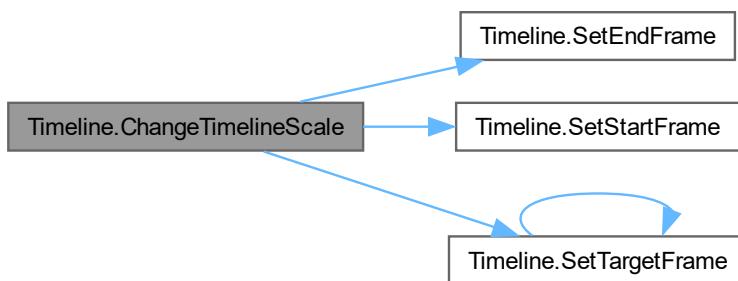
```
void Timeline.ChangeTimelineScale (   
    float delta_zoom )
```

Handles "zooming" of timeline, adjusting start and endframe of timeline as well as playhead position based on new timeline scale.

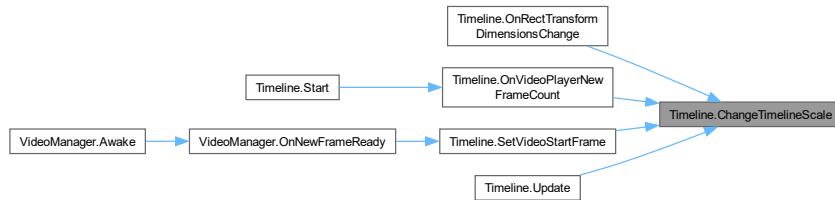
Parameters

<i>delta_zoom</i>	
-------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:

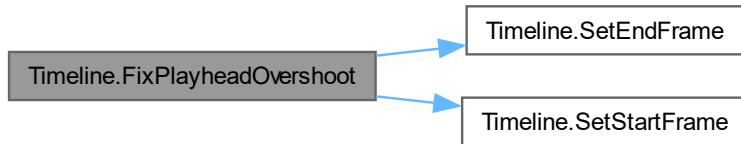


4.43.2.3 FixPlayheadOvershoot()

```
void Timeline.FixPlayheadOvershoot ( )
```

Resets playheads position back to lefthand side of the timeline.

Here is the call graph for this function:



Here is the caller graph for this function:



4.43.2.4 HoverHandoverWithChildren()

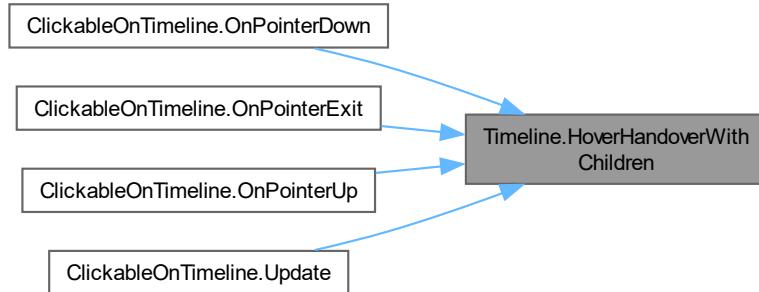
```
void Timeline.HoverHandoverWithChildren (
    GameObject child,
    bool hover )
```

Allows children of timeline to signal if timeline is hovered or not.

Parameters

<i>child</i>	signaling child.
<i>hover</i>	bool; whether or not timeline is being hovered.

Here is the caller graph for this function:

**4.43.2.5 OnNewFrameInVideoPlayer()**

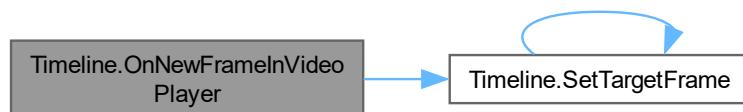
```
void Timeline.OnNewFrameInVideoPlayer (
    long newframe ) [private]
```

Callback for VideoManagers OnNewFrame event. If theres a new frame displayed by the video and the timeline isnt currently being scrubbed by user, update playhead to new frame.

Parameters

<i>newframe</i>	New frame in VideoPlayer
-----------------	--------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



4.43.2.6 OnPointerEnter()

```
void Timeline.OnPointerEnter (
    PointerEventData eventData )
```

4.43.2.7 OnPointerExit()

```
void Timeline.OnPointerExit (
    PointerEventData eventData )
```

Handles snapping, meaning when user exits timeline on left or right side of timeline the playhead should snap the start/end frame of timeline.

Parameters

<i>eventData</i>

Here is the call graph for this function:



4.43.2.8 OnRecordingToggled()

```
void Timeline.OnRecordingToggled (
    bool is_recording ) [private]
```

Callback for Eyerecorders RecordingToggled Event. Resets video and scrubber to 0 when recording starts.

Parameters

<i>is_recording</i>	Bool; True if recording is running.
---------------------	-------------------------------------

Here is the call graph for this function:

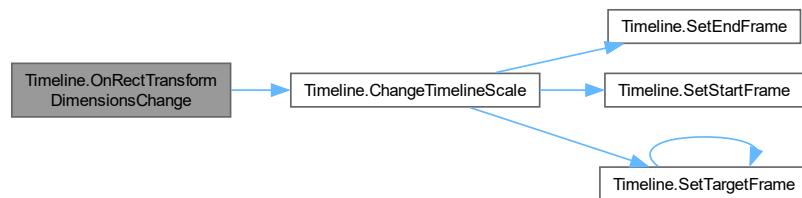


Here is the caller graph for this function:

**4.43.2.9 OnRectTransformDimensionsChange()**

```
void Timeline.OnRectTransformDimensionsChange ( ) [private]
```

Here is the call graph for this function:

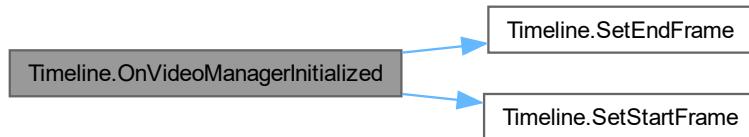


4.43.2.10 OnVideoManagerInitialized()

```
void Timeline.OnVideoManagerInitialized ( ) [private]
```

Callback for VideoManagers VideoManagerInitialized Event. When videomanager is done loading the timeline can set the first and last frame.

Here is the call graph for this function:



Here is the caller graph for this function:



4.43.2.11 OnVideoPlayerNewFrameCount()

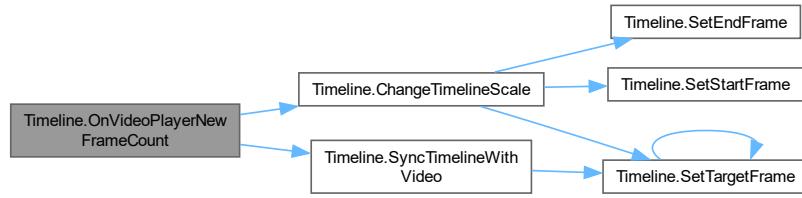
```
void Timeline.OnVideoPlayerNewFrameCount (
    ulong newframecount ) [private]
```

Callback for VideoManagers OnNewFrameCount event. The VideoPlayer might only know what exactly the last frame of the video is when playback approaches the end of the video. If the FrameCount changes, the timeline updates itself using SyncTimelineWithVideo and ChangeTimelineScale.

Parameters

<i>newframecount</i>	<input type="text"/>
----------------------	----------------------

Here is the call graph for this function:



Here is the caller graph for this function:



4.43.2.12 SetBlockEdgeSnapping()

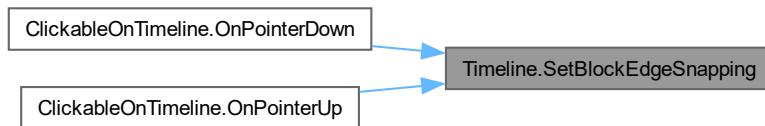
```
void Timeline.SetBlockEdgeSnapping (
    GameObject child,
    bool value )
```

Edge snapping might have to be disabled when children are currently handling input events that happen ontop of timeline.

Parameters

<code>child</code>	child that wants to disable/enable edge snapping.
<code>value</code>	bool, if true edge snapping is disabled.

Here is the caller graph for this function:



4.43.2.13 SetEndFrame()

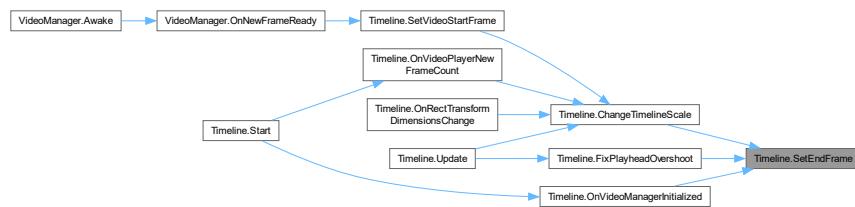
```
void Timeline.SetEndFrame (
    float frame ) [private]
```

Sets end frame of timeline and changes label at the right of timeline to display new endframe. Note: the EndFrame is not the last frame in the video, but the last frame of the video that is represented on the timeline.

Parameters

frame	<input type="text"/>
-------	----------------------

Here is the caller graph for this function:



4.43.2.14 SetStartFrame()

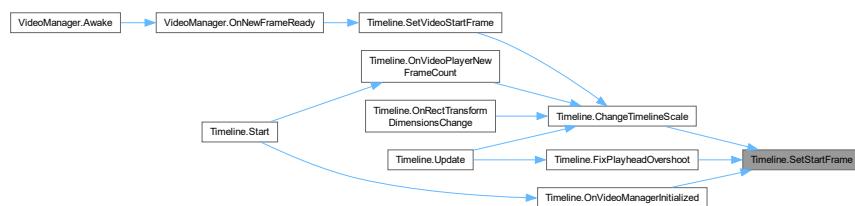
```
void Timeline.SetStartFrame (
    float frame ) [private]
```

Sets start frame of timeline and changes label at the left of timeline to display new start frame. Note: the StartFrame is not the first frame in the video, but the first frame of the video that is represented on the timeline.

Parameters

frame	<input type="text"/>
-------	----------------------

Here is the caller graph for this function:



4.43.2.15 SetTargetFrame() [1/2]

```
void Timeline.SetTargetFrame (
    long newframe,
    bool updateplayhead,
    bool updatevideo )
```

Used to update timeline to a new frame.

Parameters

<i>newframe</i>	New target frame for timeline.
<i>updateplayhead</i>	True if playhead needs to be updated.
<i>updatevideo</i>	True if video needs to be updated.

Here is the call graph for this function:

**4.43.2.16 SetTargetFrame() [2/2]**

```
void Timeline.SetTargetFrame (
    RectTransform childrect,
    bool updateplayhead,
    bool updatevideo )
```

Used by children of timeline to set target frame based on childs position on timeline.

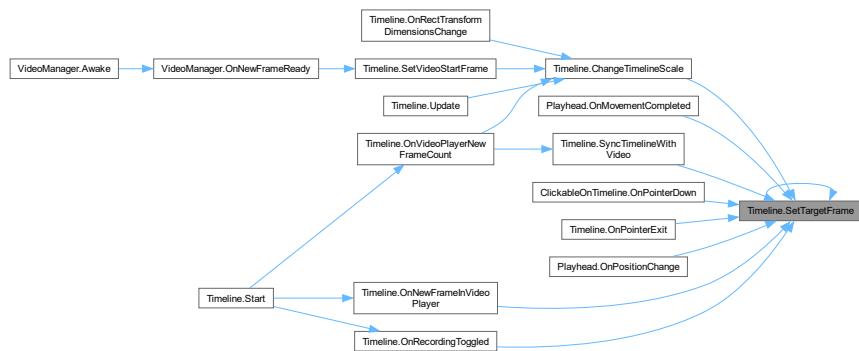
Parameters

<i>childrect</i>	Rect of child that wants to set target frame.
<i>updateplayhead</i>	bool; true if playhead needs to be updated.
<i>updatevideo</i>	bool; true if video needs to be updated.

Here is the call graph for this function:



Here is the caller graph for this function:



4.43.2.17 SetVideoStartFrame()

```
void Timeline.SetVideoStartFrame (
    float frame )
```

called by [VideoManager](#) when it has accurate information about what the first frame of the video is. Should maybe be handled by an event for better decoupling.

Parameters

<i>frame</i>	Id of first frame in video
--------------	----------------------------

Here is the call graph for this function:



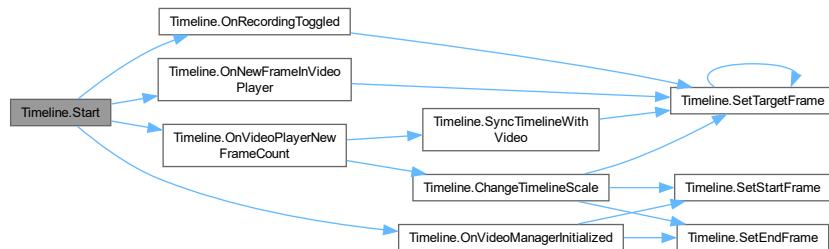
Here is the caller graph for this function:



4.43.2.18 Start()

```
void Timeline.Start ( ) [private]
```

Here is the call graph for this function:



4.43.2.19 SyncTimelineWithVideo()

```
void Timeline.SyncTimelineWithVideo ( )
```

Can be used to sync timeline target frame and playhead position to current frame of video.

Here is the call graph for this function:



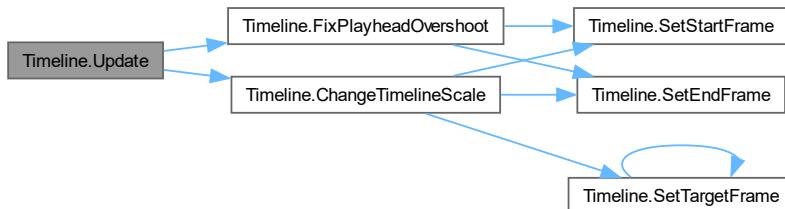
Here is the caller graph for this function:



4.43.2.20 Update()

```
void Timeline.Update ( ) [private]
```

Here is the call graph for this function:



4.43.3 Member Data Documentation

4.43.3.1 _playhead

```
Playhead Timeline._playhead [private]
```

4.43.3.2 EndFrameLabel

```
TMPRO.TextMeshProUGUI Timeline.EndFrameLabel
```

4.43.3.3 OnTargetFrameChange

```
UnityEvent<long> Timeline.OnTargetFrameChange
```

4.43.3.4 PlayheadRect

```
RectTransform Timeline.PlayheadRect
```

4.43.3.5 StartFrameLabel

```
TMP_Text Timeline.StartFrameLabel
```

4.43.4 Property Documentation

4.43.4.1 BlockedEdgeSnapping

```
bool Timeline.BlockedEdgeSnapping = false [get], [private set]
```

4.43.4.2 EndFrame

```
float Timeline.EndFrame [get], [private set]
```

4.43.4.3 Height

```
float Timeline.Height [get], [private set]
```

4.43.4.4 Instance

```
Timeline Timeline.Instance [static], [get], [private set]
```

4.43.4.5 IsHovered

```
bool Timeline.IsHovered = false [get], [private set]
```

4.43.4.6 ScrollScale

```
float Timeline.ScrollScale = 0.03f [get], [private set]
```

4.43.4.7 StartFrame

```
float Timeline.StartFrame = 0 [get], [private set]
```

4.43.4.8 TargetFrame

```
long Timeline.TargetFrame = 0 [get], [private set]
```

4.43.4.9 VideoStartFrame

```
float Timeline.VideoStartFrame = -1 [get], [private set]
```

4.43.4.10 Width

```
float Timeline.Width [get], [private set]
```

4.43.4.11 WidthPerFrame

```
float Timeline.WidthPerFrame [get], [private set]
```

4.43.4.12 ZoomFactor

```
float Timeline.ZoomFactor = 1 [get], [private set]
```

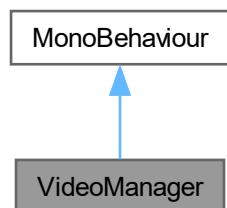
The documentation for this class was generated from the following file:

- Assets/Scripts/Video and Timeline/[Timeline.cs](#)

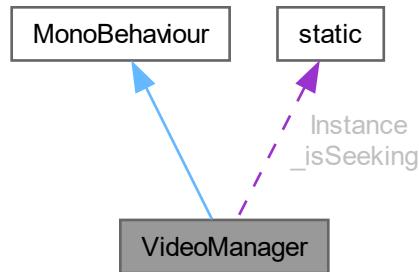
4.44 VideoManager Class Reference

This class handles the videoplayer component and its events.

Inheritance diagram for VideoManager:



Collaboration diagram for VideoManager:



Public Member Functions

- void [OnVideoLoaded](#) (string path)
Callback for when video has been loaded by videoloader. Starts preparing video player component with selected video.
- IEnumerator [SeekFrameInVideo](#) (long frame)
Function to seek frame in video. Stutters towards frame by seeking to a frame before the target frame and then stepping forward. This is done because seeking to a frame doesn't guarantee that the frame will be displayed in the video. Stuttering towards the frame makes the video display the correct frame 99% of times.
- void [PlayPause](#) ()
Toggle the play/pause state of the video when the Play/Pause button is clicked.
- void [OnVolumeSliderValueChanged](#) ()
Callback for volume slider in UI. Changes volume based on sliders value.

Public Attributes

- Slider [VolumeSlider](#)
- RenderTexture [TargetTexture](#)
- Material [SkyboxMaterial](#)

Static Public Attributes

- static [VideoManager Instance](#)

Properties

- UnityEvent< long > [OnNewFrame](#) = new() [get, private set]
- UnityEvent< ulong > [OnFrameCountChange](#) = new() [get, private set]
- UnityEvent [OnVideoManagerInitialized](#) = new() [get, private set]
- static VideoPlayer [VideoPlayer](#) [get, private set]
- static bool [IsInitialized](#) = false [get, private set]
- long [CurrentFrame](#) [get, private set]
- ulong [FrameCount](#) [get, private set]

Private Member Functions

- void [Awake \(\)](#)
- void [Start \(\)](#)
- void [OnVideoPlayerError \(VideoPlayer source, string message\)](#)
Callback for VideoPlayers errorReceived event. If VideoPlayer component encounters and error its being logged.
- void [OnVideoPlayerPrepareCompleted \(VideoPlayer source\)](#)
Callback for videoplayer components prepareCompleted event. When preparation is completed, resolution of video is read. Then a new render texture with correct resoltion is created and assigned to skybox material.
- IEnumerator [PlayUntilFrameOne \(\)](#)
Plays the video until right after being loaded. Needed to find first frame of video, because not all videos start with frame 0.
- void [OnNewFrameInTimeline \(long newframe\)](#)
Callback for timelines NewFrameInTimeline event. When timeline is targeting a new frame the video manager starts seeking that frame.
- void [Update \(\)](#)
- void [OnNewFrameReady \(VideoPlayer source, long framidx\)](#)
Callback for videoplayer newframeready event.
- void [SeekCompleted \(VideoPlayer source\)](#)
Sets _isSeeking to false when seeking is completed.
- void [OnRecordingToggled \(bool is_recording\)](#)
Callback for EyeRecorders RecordingToggled event. When recording starts start video playback. When recording ends stop video playback.

Static Private Attributes

- static bool [_isSeeking = false](#)

4.44.1 Detailed Description

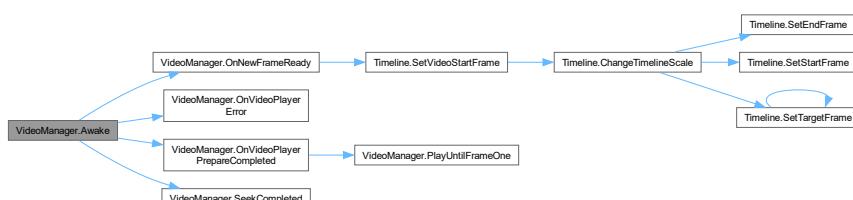
This class handles the videoplayer component and its events.

4.44.2 Member Function Documentation

4.44.2.1 Awake()

```
void VideoManager.Awake ( ) [private]
```

Here is the call graph for this function:



4.44.2.2 OnNewFrameInTimeline()

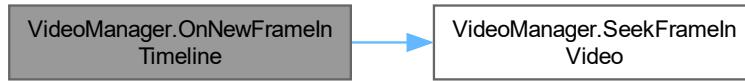
```
void VideoManager.OnNewFrameInTimeline (
    long newframe ) [private]
```

Callback for timelines NewFrameInTimeline event. When timeline is targeting a new frame the video manager starts seeking that frame.

Parameters

<i>newframe</i>	<input type="button" value=""/>
-----------------	---------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



4.44.2.3 OnNewFrameReady()

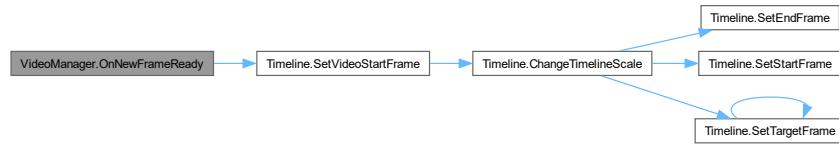
```
void VideoManager.OnNewFrameReady (
    VideoPlayer source,
    long frameIdx ) [private]
```

Callback for videoplayer newframeready event.

Parameters

<i>source</i>	<input type="button" value=""/>
<i>frameIdx</i>	<input type="button" value=""/>

Here is the call graph for this function:



Here is the caller graph for this function:



4.44.2.4 OnRecordingToggled()

```
void VideoManager.OnRecordingToggled (
    bool is_recording ) [private]
```

Callback for EyeRecorders RecordingToggled event. When recording starts start video playback. When recording ends stop video playback.

Parameters

<i>is_recording</i>	<input type="checkbox"/>
---------------------	--------------------------

Here is the caller graph for this function:



4.44.2.5 OnVideoLoaded()

```
void VideoManager.OnVideoLoaded (
    string path )
```

Callback for when video has been loaded by videoloader. Starts preparing video player component with selected video.

Parameters

<i>path</i>	
-------------	--

Here is the caller graph for this function:



4.44.2.6 OnVideoPlayerError()

```
void VideoManager.OnVideoPlayerError (
    VideoPlayer source,
    string message ) [private]
```

Callback for VideoPlayers errorReceived event. If VideoPlayer component encounters and error its being logged.

Parameters

<i>source</i>	
<i>message</i>	

Here is the caller graph for this function:



4.44.2.7 OnVideoPlayerPrepareCompleted()

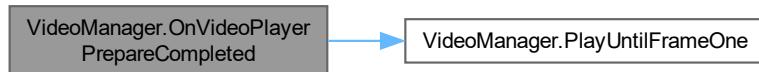
```
void VideoManager.OnVideoPlayerPrepareCompleted (
    VideoPlayer source ) [private]
```

Callback for videoplayer components prepareCompleted event. When preparation is completed, resolution of video is read. Then a new render texture with correct resoltion is created and assigned to skybox material.

Parameters

<i>source</i>	<input type="text"/>
---------------	----------------------

Here is the call graph for this function:



Here is the caller graph for this function:



4.44.2.8 OnVolumeSliderValueChanged()

```
void VideoManager.OnVolumeSliderValueChanged ( )
```

Callback for volume slider in UI. Changes volume based on sliders value.

4.44.2.9 PlayPause()

```
void VideoManager.PlayPause ( )
```

Toggle the play/pause state of the video when the Play/Pause button is clicked.

Here is the caller graph for this function:



4.44.2.10 PlayUntilFrameOne()

```
IEnumerator VideoManager.PlayUntilFrameOne ( ) [private]
```

Plays the video until right after being loaded. Needed to find first frame of video, because not all videos start with frame 0.

Returns

Here is the caller graph for this function:



4.44.2.11 SeekCompleted()

```
void VideoManager.SeekCompleted (
    VideoPlayer source ) [private]
```

Sets `_isSeeking` to false when seeking is completed.

Parameters

<code>source</code>	<input type="button" value=""/>
---------------------	---------------------------------

Here is the caller graph for this function:



4.44.2.12 SeekFrameInVideo()

```
IEnumerator VideoManager.SeekFrameInVideo (
    long frame )
```

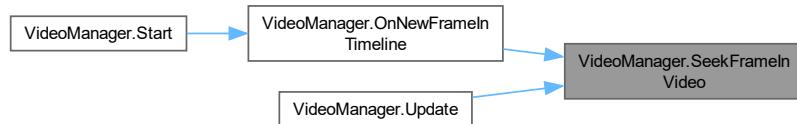
Function to seek frame in video. Stutters towards frame by seeking to a frame before the target frame and then stepping forward. This is done because seeking to a frame doesn't guarantee that the frame will be displayed in the video. Stuttering towards the frame makes the video display the correct frame 99% of times.

Parameters

<i>frame</i>	target frame
--------------	--------------

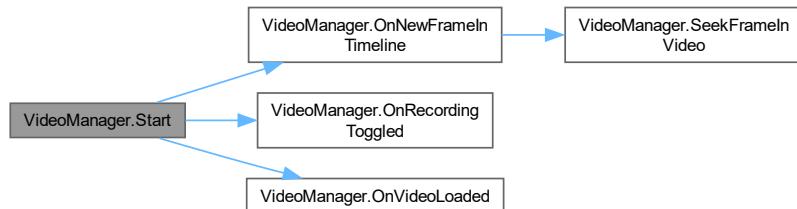
Returns

Here is the caller graph for this function:

**4.44.2.13 Start()**

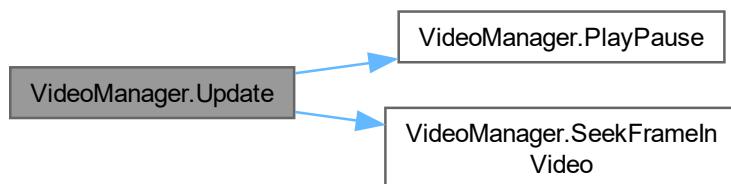
```
void VideoManager.Start ( ) [private]
```

Here is the call graph for this function:

**4.44.2.14 Update()**

```
void VideoManager.Update ( ) [private]
```

Here is the call graph for this function:



4.44.3 Member Data Documentation

4.44.3.1 `_isSeeking`

```
bool VideoManager._isSeeking = false [static], [private]
```

4.44.3.2 `Instance`

```
VideoManager VideoManager.Instance [static]
```

4.44.3.3 `SkyboxMaterial`

```
Material VideoManager.SkyboxMaterial
```

4.44.3.4 `TargetTexture`

```
RenderTexture VideoManager.TargetTexture
```

4.44.3.5 `VolumeSlider`

```
Slider VideoManager.VolumeSlider
```

4.44.4 Property Documentation

4.44.4.1 `CurrentFrame`

```
long VideoManager.CurrentFrame [get], [private set]
```

4.44.4.2 `FrameCount`

```
ulong VideoManager.FrameCount [get], [private set]
```

4.44.4.3 `IsInitialized`

```
bool VideoManager.IsInitialized = false [static], [get], [private set]
```

4.44.4.4 `OnFrameCountChange`

```
UnityEvent<ulong> VideoManager.OnFrameCountChange = new() [get], [private set]
```

4.44.4.5 OnNewFrame

```
UnityEvent<long> VideoManager.OnNewFrame = new() [get], [private set]
```

4.44.4.6 OnVideoManagerInitialized

```
UnityEvent VideoManager.OnVideoManagerInitialized = new() [get], [private set]
```

4.44.4.7 VideoPlayer

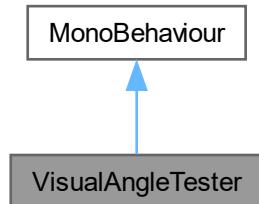
```
VideoPlayer VideoManager.VideoPlayer [static], [get], [private set]
```

The documentation for this class was generated from the following file:

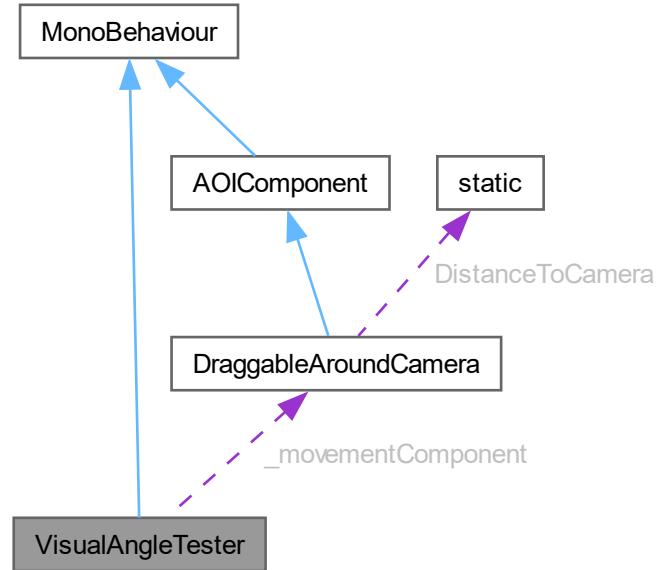
- Assets/Scripts/Video and Timeline/[VideoManager.cs](#)

4.45 VisualAngleTester Class Reference

Inheritance diagram for VisualAngleTester:



Collaboration diagram for VisualAngleTester:



Public Attributes

- `Vector3 TestDirection`
- `float phi = 1`
- `float distance = 10`
- `float size`
- `Color Color`

Private Member Functions

- `void Start ()`
- `void Update ()`
- `void OnMovementCompleted ()`

Private Attributes

- `GameObject _angle_sphere`
- `MeshRenderer _meshRenderer`
- `DraggableAroundCamera _movementComponent`

4.45.1 Member Function Documentation

4.45.1.1 OnMovementCompleted()

```
void VisualAngleTester.OnMovementCompleted ( ) [private]
```

Here is the caller graph for this function:



4.45.1.2 Start()

```
void VisualAngleTester.Start ( ) [private]
```

Here is the call graph for this function:



4.45.1.3 Update()

```
void VisualAngleTester.Update ( ) [private]
```

4.45.2 Member Data Documentation

4.45.2.1 _angle_sphere

```
GameObject VisualAngleTester._angle_sphere [private]
```

4.45.2.2 _meshRenderer

```
MeshRenderer VisualAngleTester._meshRenderer [private]
```

4.45.2.3 `_movementComponent`

```
DraggableAroundCamera VisualAngleTester._movementComponent [private]
```

4.45.2.4 `Color`

```
Color VisualAngleTester.Color
```

4.45.2.5 `distance`

```
float VisualAngleTester.distance = 10
```

4.45.2.6 `phi`

```
float VisualAngleTester.phi = 1
```

4.45.2.7 `size`

```
float VisualAngleTester.size
```

4.45.2.8 `TestDirection`

```
Vector3 VisualAngleTester.TestDirection
```

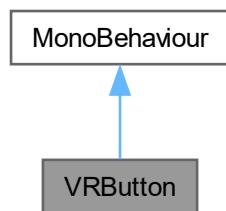
The documentation for this class was generated from the following file:

- Assets/Scripts/Application/[VisualAngleTester.cs](#)

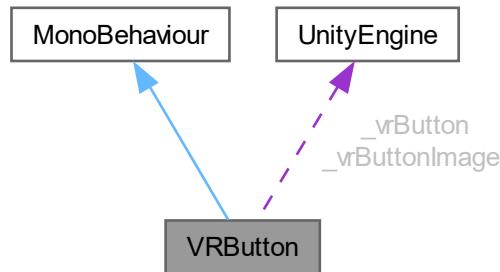
4.46 VRButton Class Reference

Attached to VR Button. Manages interactability as well as sprite changes for button.

Inheritance diagram for VRButton:



Collaboration diagram for VRButton:



Public Attributes

- Sprite `_loadingSprite`
- Sprite `_onSprite`
- Sprite `_errorSprite`
- Sprite `_offSprite`

Private Member Functions

- void `Start ()`
- void `UpdateInteractivityBasedOnRecordingStatus (bool status)`
Callback for Eyerecorders OnRecordingToggled-Event. Makes button uninteractable while recording is running.
- void `UpdateImageBasedOnVrManagerStatus (VRManager.VRStatus status)`
Callback for VRManagers OnXRStatusUpdate Event. Changes sprite color based on status of VR-Framework.

Private Attributes

- UnityEngine.UI.Button `_vrButton`
- UnityEngine.UI.Image `_vrButtonImage`

4.46.1 Detailed Description

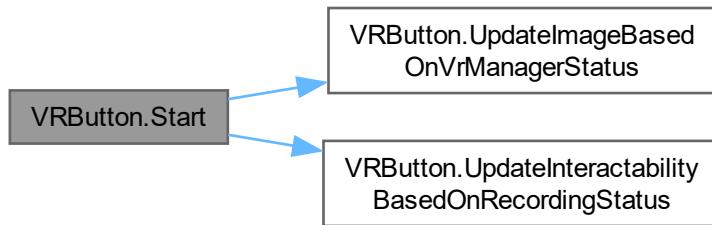
Attached to VR Button. Manages interactability as well as sprite changes for button.

4.46.2 Member Function Documentation

4.46.2.1 Start()

```
void VRButton.Start ( ) [private]
```

Here is the call graph for this function:



4.46.2.2 UpdateImageBasedOnVrManagerStatus()

```
void VRButton.UpdateImageBasedOnVrManagerStatus ( VRManager::VRStatus status ) [private]
```

Callback for VRManagers OnXRStatusUpdate Event. Changes sprite color based on status of VR-Framework.

Parameters

<code>status</code>	New status of VR-Framework
---------------------	----------------------------

Here is the caller graph for this function:



4.46.2.3 UpdateInteractabilityBasedOnRecordingStatus()

```
void VRButton.UpdateInteractabilityBasedOnRecordingStatus ( bool status ) [private]
```

Callback for Eyerecorders OnRecordingToggled-Event. Makes button uninteractable while recording is running.

Parameters

<code>status</code>	Whether or not recording is running.
---------------------	--------------------------------------

Here is the caller graph for this function:



4.46.3 Member Data Documentation

4.46.3.1 `_errorSprite`

```
Sprite VRButton._errorSprite
```

4.46.3.2 `_loadingSprite`

```
Sprite VRButton._loadingSprite
```

4.46.3.3 `_offSprite`

```
Sprite VRButton._offSprite
```

4.46.3.4 `_onSprite`

```
Sprite VRButton._onSprite
```

4.46.3.5 `_vrButton`

```
UnityEngine.UI.Button VRButton._vrButton [private]
```

4.46.3.6 `_vrButtonImage`

```
UnityEngine.UI.Image VRButton._vrButtonImage [private]
```

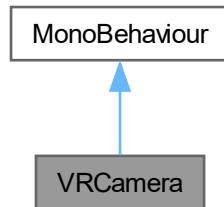
The documentation for this class was generated from the following file:

- Assets/Scripts/VR Management/[VRButton.cs](#)

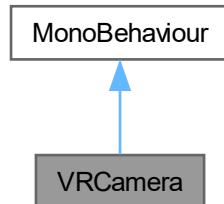
4.47 VRCamera Class Reference

Attached to the camera used by the VR headset and sets the cameras culling mask and clear flags.

Inheritance diagram for VRCamera:



Collaboration diagram for VRCamera:



Private Member Functions

- void [Awake \(\)](#)
- void [OnVRStatusChange \(VRManager.VRStatus status\)](#)
Callback for VRManagers OnXRStatusUpdate-Event. When VR is being activated checks whether or not to display Video (Skybox) or Solid Color (for TestScene)
- void [OnAOIVisibilityToggled \(bool new_visibility\)](#)
Callback for VRSettingsMenus OnAOIVisibilityToggled-Event. Changes whether or not AOIs are visible in VR.
- void [OnRecordingToggled \(bool is_recording\)](#)
Callback for Eyerecorders OnRecordingToggled-Event. Displays Skybox (Video) as soon as recording starts.
- void [OnPreVideoStimulusToggled \(bool isPreVideoTestSceneEnabled\)](#)
Callback for VRSettingsMenus OnIsPreVideoTestSceneEnabledToggled-Event. Changes Camera Clear Flags to display solid color instead of video when TestScene is enabled.

4.47.1 Detailed Description

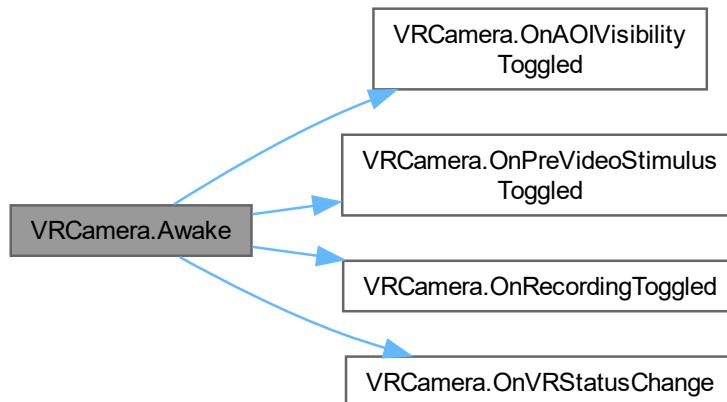
Attached to the camera used by the VR headset and sets the cameras culling mask and clear flags.

4.47.2 Member Function Documentation

4.47.2.1 Awake()

```
void VRCamera.Awake ( ) [private]
```

Here is the call graph for this function:



4.47.2.2 OnAOIVisibilityToggled()

```
void VRCamera.OnAOIVisibilityToggled (
    bool new_visibility ) [private]
```

Callback for VRSettingsMenus OnAOIVisibilityToggled-Event. Changes whether or not AOIs are visible in VR.

Parameters

<code>new_visibility</code>

Here is the caller graph for this function:



4.47.2.3 OnPreVideoStimulusToggled()

```
void VRCamera.OnPreVideoStimulusToggled (
    bool isPreVideoTestSceneEnabled ) [private]
```

Callback for VRSettingsMenus OnIsPreVideoTestSceneEnabledToggled-Event. Changes Camera Clear Flags to display solid color instead of video when TestScene is enabled.

Parameters

<i>isPreVideoTestSceneEnabled</i>	<input type="checkbox"/>
-----------------------------------	--------------------------

Here is the caller graph for this function:



4.47.2.4 OnRecordingToggled()

```
void VRCamera.OnRecordingToggled (
    bool is_recording ) [private]
```

Callback for Eyerecorders OnRecordingToggled-Event. Displays Skybox (Video) as soon as recording starts.

Parameters

<i>is_recording</i>	<input type="checkbox"/>
---------------------	--------------------------

Here is the caller graph for this function:



4.47.2.5 OnVRStatusChange()

```
void VRCamera.OnVRStatusChange (
    VRManager::VRStatus status ) [private]
```

Callback for VRManagers OnXRStatusUpdate-Event. When VR is being activated cheks whether or not to display Video (Skybox) or Solid Color (for TestScene)

Parameters

<i>status</i>	<input type="button" value=""/>
---------------	---------------------------------

Here is the caller graph for this function:



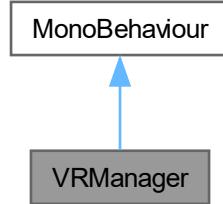
The documentation for this class was generated from the following file:

- Assets/Scripts/VR Management/[VRCamera.cs](#)

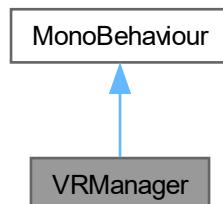
4.48 VRManager Class Reference

Handles activation and deactivation of Unitys XR Framework and SRanipal Eye Framework.

Inheritance diagram for VRManager:



Collaboration diagram for VRManager:



Public Types

- enum [VRStatus](#) { [Inactive](#) , [Loading](#) , [Error](#) , [Active](#) }

Public Member Functions

- IEnumerator [StartXRCoroutine](#) ()
Coroutine to attempt to start unity XRFramework and SRanipal Framework.
- void [ToggleXR](#) ()
function called by VRButton in UI
- void [StartXR](#) ()
Wrapper for coroutine.
- void [StopXR](#) ()
Stops Unity XR Framework as well as SRanipal Framework.

Public Attributes

- GameObject [_EyeFramework](#)
- GameObject [_XRRig](#)

Properties

- static UnityEvent< VRStatus > **OnXRStatusUpdate** = new() [get, private set]
- static VRStatus **CurrentVRStatus** [get, private set]

Private Member Functions

- void **SetCurrentVRStatusAndNotify** (VRStatus new_status)
tiny helper function to set vr status and invoke event at same time. could've beed done using auto prperties though

4.48.1 Detailed Description

Handles activation and deactivation of Unitys XR Framework and SRanipal Eye Framework.

4.48.2 Member Enumeration Documentation

4.48.2.1 VRStatus

enum VRManager.VRStatus

Enumerator

Inactive	
Loading	
Error	
Active	

4.48.3 Member Function Documentation

4.48.3.1 SetCurrentVRStatusAndNotify()

```
void VRManager.SetCurrentVRStatusAndNotify (
    VRStatus new_status ) [private]
```

tiny helper function to set vr status and invoke event at same time. could've beed done using auto prperties though

Parameters

<i>new_status</i>	<input type="text"/>
-------------------	----------------------

Here is the caller graph for this function:



4.48.3.2 StartXR()

```
void VRManager.StartXR ( )
```

Wrapper for coroutine.

Here is the call graph for this function:



Here is the caller graph for this function:



4.48.3.3 StartXRCoroutine()

```
IEnumerator VRManager.StartXRCoroutine ( )
```

Coroutine to attempt to start unity XRFramework and SRanipal Framework.

Returns

Here is the call graph for this function:



Here is the caller graph for this function:



4.48.3.4 StopXR()

```
void VRManager.StopXR( )
```

Stops Unity XR Framework as well as SRanipal Framework.

Here is the call graph for this function:



Here is the caller graph for this function:



4.48.3.5 ToggleXR()

```
void VRManager.ToggleXR ( )
```

function called by [VRButton](#) in UI

Here is the call graph for this function:



4.48.4 Member Data Documentation

4.48.4.1 _EyeFramework

```
GameObject VRManager._EyeFramework
```

4.48.4.2 _XRRig

```
GameObject VRManager._XRRig
```

4.48.5 Property Documentation

4.48.5.1 CurrentVRStatus

```
VRStatus VRManager.CurrentVRStatus [static], [get], [private set]
```

4.48.5.2 OnXRStatusUpdate

```
UnityEvent<VRStatus> VRManager.OnXRStatusUpdate = new() [static], [get], [private set]
```

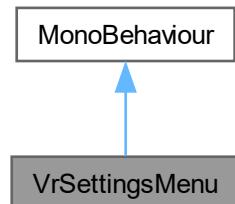
The documentation for this class was generated from the following file:

- Assets/Scripts/VR Management/[VRManager.cs](#)

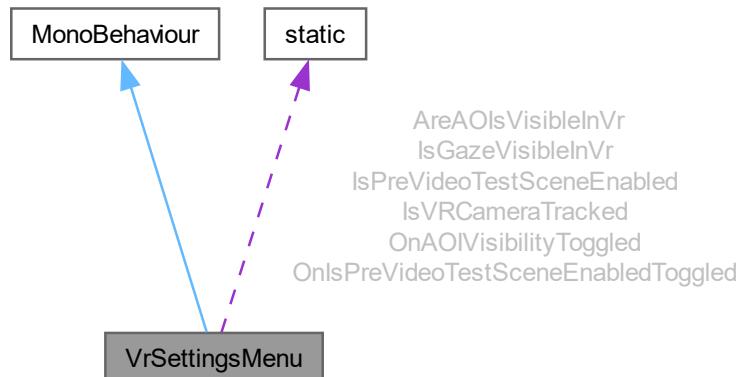
4.49 VrSettingsMenu Class Reference

Attached to UI GameObject "TopBar". Handles all user input related to the VR Settings Menu and invokes events to notify other Scripts of setting-changes.

Inheritance diagram for VrSettingsMenu:



Collaboration diagram for VrSettingsMenu:



Public Member Functions

- void [ToggleVisibility \(\)](#)
Display or hide settings menu.
- void [ToggleVRCameraTracking \(\)](#)
Called from UI. Sets static property whenever toggle is used. Property is read by [EditorCamera](#).
- void [TogglePreVideoTestScene \(\)](#)
Called from UI.
- void [ToggleAOIVisibilityInVr \(\)](#)
Called from UI.

- void [ToggleGazeVisibilityInVr \(\)](#)
Called from UI.
- void [OnNewFixationDurationThresholdEntered \(\)](#)
Called from UI.
- void [OnNewFixationDispersionThresholdEntered \(\)](#)
Called from UI.
- void [OnNewFixationVelocityThresholdEntered \(\)](#)
Called from UI.
- void [OnNewDwellToleranceEntered \(\)](#)
Called from UI.
- void [LaunchCalibration \(\)](#)
Called from UI. Launches SRanipal Calibration.

Public Attributes

- GameObject [menu](#)
- Toggle [CameraTrackingToggle](#)
- Toggle [PreVideoTestSceneToggle](#)
- Toggle [AOIVisibilityToggle](#)
- Toggle [GazeVisibilityToggle](#)
- TMP.TMP_InputField [FixationDurationThresholdInput](#)
- TMP.TMP_InputField [FixationDispersionThresholdInput](#)
- TMP.TMP_InputField [FixationVelocityThresholdInput](#)
- TMP.TMP_InputField [DwellToleranceInput](#)
- GameObject [EyeFramework](#)

Static Public Attributes

- static bool [IsVRCameraTracked](#)
- static bool [IsPreVideoTestSceneEnabled](#)
- static bool [AreAOIsVisibleInVr](#)
- static bool [IsGazeVisibleInVr](#)
- static UnityEvent< bool > [OnAOIVisibilityToggled](#) = new()
- static UnityEvent< bool > [OnIsPreVideoTestSceneEnabledToggled](#) = new()

Private Member Functions

- void [Start \(\)](#)
- void [Update \(\)](#)

4.49.1 Detailed Description

Attached to UI GameObject "TopBar". Handles all user input related to the VR Settings Menu and invokes events to notify other Scripts of setting-changes.

4.49.2 Member Function Documentation

4.49.2.1 LaunchCalibration()

```
void VrSettingsMenu.LaunchCalibration ( )
```

Called from UI. Launches SRanipal Calibration.

Here is the caller graph for this function:



4.49.2.2 OnNewDwellToleranceEntered()

```
void VrSettingsMenu.OnNewDwellToleranceEntered ( )
```

Called from UI.

4.49.2.3 OnNewFixationDispersionThresholdEntered()

```
void VrSettingsMenu.OnNewFixationDispersionThresholdEntered ( )
```

Called from UI.

4.49.2.4 OnNewFixationDurationThresholdEntered()

```
void VrSettingsMenu.OnNewFixationDurationThresholdEntered ( )
```

Called from UI.

Here is the call graph for this function:



4.49.2.5 OnNewFixationVelocityThresholdEntered()

```
void VrSettingsMenu.OnNewFixationVelocityThresholdEntered ( )
```

Called from UI.

4.49.2.6 Start()

```
void VrSettingsMenu.Start ( ) [private]
```

4.49.2.7 ToggleAOIVisibilityInVr()

```
void VrSettingsMenu.ToggleAOIVisibilityInVr ( )
```

Called from UI.

4.49.2.8 ToggleGazeVisibilityInVr()

```
void VrSettingsMenu.ToggleGazeVisibilityInVr ( )
```

Called from UI.

Here is the caller graph for this function:

**4.49.2.9 TogglePreVideoTestScene()**

```
void VrSettingsMenu.TogglePreVideoTestScene ( )
```

Called from UI.

4.49.2.10 ToggleVisibility()

```
void VrSettingsMenu.ToggleVisibility ( )
```

Display or hide settings menu.

4.49.2.11 ToggleVRCameraTracking()

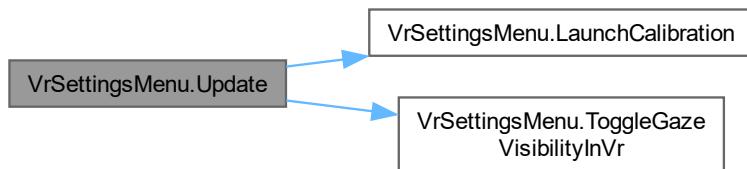
```
void VrSettingsMenu.ToggleVRCameraTracking ( )
```

Called from UI. Sets static property whenever toggle is used. Property is read by [EditorCamera](#).

4.49.2.12 Update()

```
void VrSettingsMenu.Update ( ) [private]
```

Here is the call graph for this function:



4.49.3 Member Data Documentation

4.49.3.1 AOIVisibilityToggle

```
Toggle VrSettingsMenu.AOIVisibilityToggle
```

4.49.3.2 AreAOIsVisibleInVr

```
bool VrSettingsMenu.AreAOIsVisibleInVr [static]
```

4.49.3.3 CameraTrackingToggle

```
Toggle VrSettingsMenu.CameraTrackingToggle
```

4.49.3.4 DwellToleranceInput

```
TMP TMP_InputField VrSettingsMenu.DwellToleranceInput
```

4.49.3.5 EyeFramework

```
GameObject VrSettingsMenu.EyeFramework
```

4.49.3.6 FixationDispersionThresholdInput

```
TMP TMP_InputField VrSettingsMenu.FixationDispersionThresholdInput
```

4.49.3.7 FixationDurationThresholdInput

```
TMP TMP_InputField VrSettingsMenu.FixationDurationThresholdInput
```

4.49.3.8 FixationVelocityThresholdInput

```
TMP TMP_InputField VrSettingsMenu.FixationVelocityThresholdInput
```

4.49.3.9 GazeVisibilityToggle

```
Toggle VrSettingsMenu.GazeVisibilityToggle
```

4.49.3.10 IsGazeVisibleInVr

```
bool VrSettingsMenu.IsGazeVisibleInVr [static]
```

4.49.3.11 IsPreVideoTestSceneEnabled

```
bool VrSettingsMenu.IsPreVideoTestSceneEnabled [static]
```

4.49.3.12 IsVRCameraTracked

```
bool VrSettingsMenu.IsVRCameraTracked [static]
```

4.49.3.13 menu

```
GameObject VrSettingsMenu.menu
```

4.49.3.14 OnAOIVisibilityToggled

```
UnityEvent<bool> VrSettingsMenu.OnAOIVisibilityToggled = new() [static]
```

4.49.3.15 OnIsPreVideoTestSceneEnabledToggled

```
UnityEvent<bool> VrSettingsMenu.OnIsPreVideoTestSceneEnabledToggled = new() [static]
```

4.49.3.16 PreVideoTestSceneToggle

```
Toggle VrSettingsMenu.PreVideoTestSceneToggle
```

The documentation for this class was generated from the following file:

- Assets/Scripts/VR Management/[VrSettingsMenu.cs](#)

Chapter 5

File Documentation

5.1 Assets/Scripts/AOI Management/AOI Components/Animation.cs File Reference

Classes

- class [Animation](#)

This script does a lot of things and should probably be refactored into multiple classes. a) Creates & Updates Keyframes for position, size and shape of AOI Does this by listening to DraggableAroundCameras and Resizables events b) Calculates position, size and shape for Frames between Keyframes using interpolation c) Animates object based on current frame of video by setting position, size and shape according to calculated frames.

- class [Animation.FrameData](#)

5.2 Assets/Scripts/AOI Management/AOI Components/AOIComponent.cs File Reference

Classes

- class [AOIComponent](#)

This is the base class for components that make up an area of interest. The user can highlight and work with one AOI at a time, marking that one as active (done by the AOI Manager) This class subscribes to the event of changing the active AOI and specifies the relevant methods (OnActivate, OnDeactivate) that need to be implemented by derived classes.

5.3 Assets/Scripts/AOI Management/AOI Components/Colorable.cs File Reference

Classes

- class [Colorable](#)

Class that was supposed to allow AOIs to be recolored from AOI List or RightClickMenu. Not implemented though, so class is mostly useless. Currently only handles color of AOI based on whether or not AOI is selected by user.

5.4 Assets/Scripts/AOI Management/AOI Components/DraggableAroundCamera.cs File Reference

Classes

- class [DraggableAroundCamera](#)

This script is attached to AOIs and allows them to be dragged around to position them within the 360 degree video.

5.5 Assets/Scripts/AOI Management/AOI Components/Listable.cs File Reference

Classes

- class [Listable](#)

This script manages the name of the AOI and its representation in the AOIList in the UI.

5.6 Assets/Scripts/AOI Management/AOI Components/Resizable.cs File Reference

Classes

- class [Resizable](#)

This script is attached to AOIs and is responsible for the creation of handles as well as resizing. Resizing happens when the user drags a handle of the currently selected AOI across the screen. Something of note: Resizing is done by changing the vertices of the AOIs mesh. Moving the vertices doesn't automatically recalculate the center (origin) of the game object, so this class manually recenters the objects origin during and after resizing is completed.

5.7 Assets/Scripts/AOI Management/AOI Handles/DraggableAlongXYPlane.cs File Reference

Classes

- class [DraggableAlongXYPlane](#)

this script is attached to every AOI handle and allows the handle to be dragged along the XY plane of its parent(the AOI)

5.8 Assets/Scripts/AOI Management/AOI Handles/HandleForResizable.cs File Reference

Classes

- class [HandleForResizable](#)

This script is attached to every AOI-Handle. It signals the parent object that a handle has been clicked and resizing should start. Requires the parent to have the "Resizable" Component.

5.9 Assets/Scripts/AOI Management/AOIManager.cs File Reference

Classes

- class [AOIManager](#)

This class does 3 things 1.Manages AOIs based on user input a) Deactivates currently active AOIs and spawns new AOIs if user clicks on empty space in 360 degree video b) Sets an AOI as active when user clicks on AOI in 360 degree video and deactivates the previously selected AOI.

5.10 Assets/Scripts/AOI Management/CameraDrag.cs File Reference

Classes

- class [CameraDrag](#)

This script is attached to the Editor Camera and allows the User to change the portion of the 360 degree video being displayed, by holding down right click and dragging. The camera then rotates based on the cursor movement.

5.11 Assets/Scripts/AOI Management/EditorCamera.cs File Reference

Classes

- class [EditorCamera](#)

This script is attached to the editor camera and does two things a) gives a static reference to the [EditorCamera](#) to be used for e.g. position calculation etc. b) matches the editor camera rotation to the VR cameras rotation if flag is set by user.

5.12 Assets/Scripts/Application/FPSCounter.cs File Reference

Classes

- class [FPSCounter](#)

Script to display FPS counter on the top right of screen. Source: <https://forum.unity.com/threads/fps-counter.505495/>.

5.13 Assets/Scripts/Application/LogMessagesManager.cs File Reference

Classes

- class [LogMessagesManager](#)

Script managing the creation of logs in the error/debug log console. Console can be opened and closed by pressing 'F1'.

5.14 Assets/Scripts/Application/MenuCloseLogic.cs File Reference

Classes

- class [MenuCloseLogic](#)

*Tiny little script that's attached to everything that should be "closed" when the user clicks outside of it::s bounds.
Mostly used for menus that should disappear when user clicks outside of them.*

5.15 Assets/Scripts/Application/RightClickMenu.cs File Reference

Classes

- class [RightClickMenu](#)

This script can be attached to any gameobject to display a menu when the gameobject is right-clicked. Currently used on AOIs and keyframe-icons.

5.16 Assets/Scripts/Application/Setup.cs File Reference

Classes

- class [Setup](#)

Currently doesn't do too much. Sets target framerate of application to 120Hz.

5.17 Assets/Scripts/Application/VisualAngleTester.cs File Reference

Classes

- class [VisualAngleTester](#)

5.18 Assets/Scripts/Eyetracking and Gaze Rendering/ChangeColorOnDwell.cs File Reference

Classes

- class [ChangeColorOnDwell](#)

This script inherits from the [RaycastHitHandler](#) and therefore its functions are called by the [GazeRaycaster](#). It simply changes the color of the object it is attached to when the gaze of the person wearing the vr headset is over the object.

5.19 Assets/Scripts/Eyetracking and Gaze Rendering/DebugReRun.cs File Reference

Classes

- class [DebugReRun](#)

5.20 Assets/Scripts/Eyetracking and Gaze Rendering/EventDetection.cs File Reference

Classes

- class [EventDetection](#)

This script contains all the event-detection algorithms for detecting dwells and fixations with their respective parameters like duration etc.

- class [EventDetection.Event](#)
- class [EventDetection.AoiParameters](#)
- class [Helper](#)

5.21 Assets/Scripts/Eyetracking and Gaze Rendering/EyeRecorder.cs File Reference

Classes

- class [EyeRecorder](#)

Derived from SRanipal example script. This script is responsible for getting the eyedata from the VR-Headset, storing the data during recording as well as carrying out the raycast to check whether or not the gaze is hitting a AOI. Note: Callback runs on a separate thread to report at ~120hz. Unity is not threadsafe and cannot call any UnityEngine api from within callback thread.

- class [EyeRecorder.MonoPInvokeCallbackAttribute](#)

Required class for IL2CPP scripting backend support.

5.22 Assets/Scripts/Eyetracking and Gaze Rendering/EyeTrackingDataSample.cs File Reference

Classes

- class [EyeTrackingDataSample](#)

Dataclass for storing all relevant data for a single eyetracking sample.

5.23 Assets/Scripts/Eyetracking and Gaze Rendering/GazeRaycaster.cs File Reference

Classes

- class [GazeRaycaster](#)

Creates a raycast from current eyedata gaze direction. If an object with the [RaycastHitHandler](#) component is hit by racast this script calls the [OnRaycastEnter](#) and [OnRaycastExit](#) methods on entry and exit of the object. This allows objects to react to being looked at by the participant.

5.24 Assets/Scripts/Eyetracking and Gaze Rendering/GazeRayRenderer.cs File Reference

Classes

- class [GazeRayRenderer](#)

Derived from SRanipal Example Scripts. This script displays the current gaze of the participant as a colored ray within the experimentator-view of the application.

5.25 Assets/Scripts/Eyetracking and Gaze Rendering/RaycastHitHandler.cs File Reference

Classes

- class [RaycastHitHandler](#)

Abstract class that defines OnRaycastEnter and OnRaycastExit for interaction with [GazeRaycaster](#). Should probably be an interface in the future.

5.26 Assets/Scripts/Eyetracking and Gaze Rendering/RecordButton.cs File Reference

Classes

- class [RecordButton](#)

Script attached to record button at the top right corner of UI. Handles sprite changes based on events.

5.27 Assets/Scripts/Eyetracking and Gaze Rendering/Simulation/EventDetectionTester.cs File Reference

Classes

- class [EventDetectionTester](#)

Some testing code for the event detection. Generates a list EyeTrackingDataSamples, then creates an event detection object and passes the list as parameter. The created csv file can be checked manually and compared to the created EyeTrackingDataSamples.

- class [EventDetectionTester.TesterSample](#)

5.28 Assets/Scripts/Eyetracking and Gaze Rendering/Simulation/GazeSimulator.cs File Reference

Classes

- class [GazeSimulator](#)

Another test-script. The code generates fixations at a specified Vector3. Instead of creating a eventdetection object here the fake_data is used in the eyerecorder class when the UseSimulatedDataInstead flag is set to true.

5.29 Assets/Scripts/Logging and Visualization/EyeDataLogger.cs File Reference

5.30 Assets/Scripts/Logging and Visualization/GazeToTextureMapper.cs File Reference

Classes

- class [GazeToTextureMapper](#)

Proof of concept class. Derived from shader code of Unity PanoramicShader. Allows to map the gaze direction onto pixel coordinates of the skybox-texture. Can be used in the future to create different types of visualizations.

5.31 Assets/Scripts/Logging and Visualization/SequenceChartPythonProcess.cs File Reference

Classes

- class [SequenceChartPythonProcess](#)

Calls python process that creates sequence chart from filteredData csv file.

5.32 Assets/Scripts/Project Management/ProjectManager.cs File Reference

5.33 Assets/Scripts/Project Management/SaveAndLoadAOIs.cs File Reference

Classes

- class [SaveAndLoadAOIs](#)
Handles saving and loading AOIs to and from a json file.
- class [SaveAndLoadAOIs.AOISaveData](#)
- class [SaveAndLoadAOIs.Datalist](#)

5.34 Assets/Scripts/Video and Timeline/ClickableOnTimeline.cs File Reference

Classes

- class [ClickableOnTimeline](#)

This class serves as a base for all clickable children of the timeline. it makes sure that the hover state between the timeline and its children is properly handed-over. For example if the user clicks on a keyframe and then drags the cursor, that should not move the playhead on the timeline because the user wanted to interact with the keyframe, not the timeline. Vice versa if the user scrubs the playhead across the timeline hovering over a keyframe should not trigger a click on the keyframe.

5.35 Assets/Scripts/Video and Timeline/DisplayAsFrameOnTimeline.cs File Reference

Classes

- class [DisplayAsFrameOnTimeline](#)

This classed is used for any UI element that should be displayed as a certain frame on the timeline for example a keyframe for frame 47 always needs to be positioned where frame 47 is on the timeline, regardless of scale / zoom level of the timeline. This class handles that.

5.36 Assets/Scripts/Video and Timeline/DraggableOnTimeline.cs File Reference

Classes

- class [DraggableOnTimeline](#)

This class allows an object which is displayed on the timeline (using `DisplayAsFrameOnTimelineCOnponent`) to be dragged along the timeline using the mouse. (Used by the playhead to scrub along the timeline)

5.37 Assets/Scripts/Video and Timeline/Playhead.cs File Reference

Classes

- class [Playhead](#)

Implements the playhead. Inherits [DraggableOnTimeline](#).

5.38 Assets/Scripts/Video and Timeline/Timeline.cs File Reference

Classes

- class [Timeline](#)

Implements functionality of timeline. Allows timeline to be scrubbed and to zoom in on the timeline to be able to move playhead with more precision to better place keyframes.

5.39 Assets/Scripts/Video and Timeline/VideoLoader.cs File Reference

5.40 Assets/Scripts/Video and Timeline/VideoManager.cs File Reference

Classes

- class [VideoManager](#)

This class handles the videoplayer component and its events.

5.41 Assets/Scripts/VR Management/PreVideoTestScene.cs File Reference

Classes

- class [PreVideoTestScene](#)

Handles display of testscene after vr is enabled but before recording starts.

5.42 Assets/Scripts/VR Management/VRButton.cs File Reference

Classes

- class [VRButton](#)

Attached to VR Button. Manages interactability as well as sprite changes for button.

5.43 Assets/Scripts/VR Management/VRCamera.cs File Reference

Classes

- class [VRCamera](#)

Attached to the camera used by the VR headset and sets the cameras culling mask and clear flags.

5.44 Assets/Scripts/VR Management/VRManager.cs File Reference

Classes

- class [VRManager](#)

Handles activation and deactivation of Unitys XR Framework and SRanipal Eye Framework.

5.45 Assets/Scripts/VR Management/VrSettingsMenu.cs File Reference

Classes

- class [VrSettingsMenu](#)

Attached to UI GameObject "TopBar". Handles all user input related to the VR Settings Menu and invokes events to notify other Scripts of setting-changes.

Index

- _EyeFramework
 - VRManager, 205
- _GazeRayRenderer
 - GazeRayRenderer, 108
- _LengthOfRay
 - GazeRayRenderer, 108
- _X
 - CameraDrag, 40
- _XR Rig
 - VRManager, 205
- _Y
 - CameraDrag, 40
- _angle_sphere
 - VisualAngleTester, 192
- _averageCounter
 - FPSCounter, 99
- _averageFromAmount
 - FPSCounter, 99
- _cacheNumbersAmount
 - FPSCounter, 99
- _canvas
 - AOI Manager, 35
 - CameraDrag, 40
- _collider
 - Resizable, 148
- _color
 - Colorable, 53
- _contentContainer
 - Listable, 122
 - RightClickMenu, 155
- _currentAveraged
 - FPSCounter, 99
- _errorSprite
 - VRButton, 196
- _eye_callback_registered
 - EyeRecorder, 94
- _frameRateSamples
 - FPSCounter, 100
- _gazeSphere
 - GazeRayRenderer, 108
- _gazeSphereRenderer
 - GazeRayRenderer, 108
- _handles
 - Resizable, 148
- _inputField
 - Listable, 122
- _isSeeking
 - VideoManager, 189
- _itemInstance
 - Listable, 122
 - RightClickMenu, 155
- _loadingSprite
 - VRButton, 196
- _manuallyTriggeredMovementDueToOverlap
 - AOI Manager, 35
- _manuallyTriggeredResizeDueToOverlap
 - AOI Manager, 35
- _menuPrefab
 - RightClickMenu, 155
- _meshRenderer
 - VisualAngleTester, 192
- _mouseOffset
 - DraggableOnTimeline, 73
- _movementComponent
 - Animation, 23
 - VisualAngleTester, 192
- _objectIsBeingChanged
 - Animation, 23
- _objectToManuallyResize
 - AOI Manager, 35
- _offSprite
 - RecordButton, 140
 - VRButton, 196
- _offset
 - DraggableAlongXYPlane, 63
 - DraggableAroundCamera, 67
- _onSprite
 - RecordButton, 140
 - VRButton, 196
- _playhead
 - Timeline, 178
- _ray_color
 - GazeRayRenderer, 108
- _recButton
 - RecordButton, 140
- _recButtonImage
 - RecordButton, 140
- _recordedEyeData
 - EyeRecorder, 94
- _rect
 - ClickableOnTimeline, 48
- _renderer
 - ChangeColorOnDwell, 43
 - Colorable, 53
 - Resizable, 148
- _resizeComponent
 - Animation, 23

_timestampOffset
 EyeRecorder, 95

_vertices
 Resizable, 148

_vrButton
 VRButton, 196

_vrButtonImage
 VRButton, 196

ACTIVE
 Colorable, 50

Active
 VRManager, 202

ActiveAOI
 AOIManager, 35

ActiveAOIchanged
 AOIComponent, 25

AddDeleteOnStart
 RightClickMenu, 155

AddItem
 RightClickMenu, 151

Animation, 9
 _movementComponent, 23
 _objectIsBeingChanged, 23
 _resizeComponent, 23
 Awake, 12
 CalculateInterpolatedFrames, 12
 CreateKeyframe, 13
 DeleteKeyframeFromRightClickMenu, 13
 GetKeyframeIndexByFrame, 14
 HideKeyframes, 14
 InterpolatedFrames, 23
 IsFrameInKeyframes, 15
 IsObjectInCorrectState, 15
 KeyframePrefab, 23
 Keyframes, 23
 LoadKeyframes, 16
 ObjectChangeStarted, 16
 OnActivate, 17
 OnDeactivate, 17
 OnDestroy, 17
 SetKeyframeVisibility, 18
 SetObjectsStateToFrameData, 18
 SetupRightClickMenu, 19
 ShowKeyframes, 20
 Start, 20
 ToggleKeyframeVisibilityFromRightClickMenu, 20
 Update, 21
 UpdateKeyframe, 21
 UpdateOrCreateKeyframeOnObjectChangeCompleted, 22

Animation.FrameData, 100
 Position, 100
 Rotation, 100
 Ulrepresentation, 100
 Vertices, 100
 VideoFrame, 101
 Visibility, 101

aoi_test_selector

EventDetectionTester.TesterSample, 164

AoiColors
 Colorable, 50

AOIComponent, 24
 ActiveAOIchanged, 25
 OnActivate, 25
 OnDeactivate, 26
 OnDestroy, 26
 Start, 26

AOIManager, 27
 _canvas, 35
 _manuallyTriggeredMovementDueToOverlap, 35
 _manuallyTriggeredResizeDueToOverlap, 35
 _objectToManuallyResize, 35
 ActiveAOI, 35
 AOIPrefab, 35
 Awake, 29
 CompleteManualResizingDueToOverlap, 29
 FlipMesh, 35
 Instance, 35
 NewActiveAOI, 36
 OnTestSceneToggled, 30
 OnVrStatusChange, 30
 SetActiveObject, 31
 SpawnAoi, 32
 SpawnAOlatCursorPosition, 32
 SpawnDistance, 36
 Start, 33
 StartManualResizingDueToOverlap, 33
 ToggleAOIColliders, 34
 Update, 34

AoiName
 EventDetection.AoiParameters, 37
 EventDetection.Event, 78
 EyeTrackingDataSample, 97

AoiParameters
 EventDetection.AoiParameters, 36

AOIPrefab
 AOIManager, 35

AOIVisibilityToggle
 VrSettingsMenu, 210

AreAOIsVisibleInVr
 VrSettingsMenu, 210

Assets/Scripts/AOI Management/AOI Components/Animation.cs, 213

Assets/Scripts/AOI Management/AOI Components/AOIComponent.cs, 213

Assets/Scripts/AOI Management/AOI Components/Colorable.cs, 213

Assets/Scripts/AOI Management/AOI Components/DraggableAroundCamera.cs, 214

Assets/Scripts/AOI Management/AOI Components/Listable.cs, 214

Assets/Scripts/AOI Management/AOI Components/Resizable.cs, 214

Assets/Scripts/AOI Management/AOI Handles/DraggableAlongXYPlane.cs, 214

Assets/Scripts/AOI Management/AOI Handles/HandleForRays.cs, 214
Assets/Scripts/AOI Management/AOIManager.cs, 215
Assets/Scripts/AOI Management/CameraDrag.cs, 215
Assets/Scripts/AOI Management/EditorCamera.cs, 215
Assets/Scripts/Application/FPSCounter.cs, 215
Assets/Scripts/Application/LogMessagesManager.cs, 215
Assets/Scripts/Application/MenuCloseLogic.cs, 216
Assets/Scripts/Application/RightClickMenu.cs, 216
Assets/Scripts/Application/Setup.cs, 216
Assets/Scripts/Application/VisualAngleTester.cs, 216
Assets/Scripts/Eyetracking and Gaze Rendering/ChangeColorOnDwell.cs, 216
Assets/Scripts/Eyetracking and Gaze Rendering/DebugReRun.cs, 216
Assets/Scripts/Eyetracking and Gaze Rendering/EventDetection.cs, 217
Assets/Scripts/Eyetracking and Gaze Rendering/EyeRecorder.cs, 217
Assets/Scripts/Eyetracking and Gaze Rendering/EyeTracking.cs, 217
Assets/Scripts/Eyetracking and Gaze Rendering/GazeRaycaster.cs, 217
Assets/Scripts/Eyetracking and Gaze Rendering/GazeRayRenderer.cs, 218
Assets/Scripts/Eyetracking and Gaze Rendering/RaycastHitHandler.cs, 218
Assets/Scripts/Eyetracking and Gaze Rendering/RecordButton.cs, 218
Assets/Scripts/Eyetracking and Gaze Rendering/Simulation/EventDetectionTester.cs, 218
Assets/Scripts/Eyetracking and Gaze Rendering/Simulation/GazeSimulator.cs, 218
Assets/Scripts/Logging and Visualization/EyeDataLogger.cs, 219
Assets/Scripts/Logging and Visualization/GazeToTextureMapper.cs, 219
Assets/Scripts/Logging and Visualization/SequenceChartPythonProcess.cs, 219
Assets/Scripts/Project Management/ProjectManager.cs, 219
Assets/Scripts/Project Management/SaveAndLoadAOIs.cs, 219
Assets/Scripts/Video and Timeline/ClickableOnTimeline.cs, 219
Assets/Scripts/Video and Timeline/DisplayAsFrameOnTimeline.cs, 220
Assets/Scripts/Video and Timeline/DraggableOnTimeline.cs, 220
Assets/Scripts/Video and Timeline/Playhead.cs, 220
Assets/Scripts/Video and Timeline/Timeline.cs, 220
Assets/Scripts/Video and Timeline/VideoLoader.cs, 220
Assets/Scripts/Video and Timeline/VideoManager.cs, 220
Assets/Scripts/VR Management/PreVideoTestScene.cs, 221
Assets/Scripts/VR Management/VRButton.cs, 221
Assets/Scripts/VR Management/VRManagement.cs, 221
Assets/Scripts/VR Management/VRCamera.cs, 221
Assets/Scripts/VR Management/VRManager.cs, 221
Assets/Scripts/VR Management/VrSettingsMenu.cs, 221
Awake
 Animation, 12
 AOIManager, 29
 FPSCounter, 99
 GazeRaycaster, 102
 GazeRayRenderer, 106
 LogMessagesManager, 125
 PreVideoTestScene, 134
 RightClickMenu, 151
 Timeline, 167
 VideoManager, 182
 VRCamera, 198
BlockedEdgeSnapping
CachedNumberStrings
 DataSample.cs, 100
 FPSCounter, 100
CalculateInterpolatedFrames
 Animation, 12
CallSequenceChartScript
 SequenceChartPythonProcess, 159
Camera
 CachedNumberStrings
 DataSample.cs, 100
 FPSCounter, 100
 CalculateInterpolatedFrames
 Animation, 12
 CallSequenceChartScript
 SequenceChartPythonProcess, 159
 Camera
 EditorCamera, 76
 CameraDrag, 38
 _X, 40
 _Y, 40
 _canvas, 40
 _Speed, 40
 GazeSimulator.cs
 _Start, 39
 _Update, 39
 CameraTrackingToggle
 VrSettingsMenu, 210
Canvas
 Canvas
 DraggableOnTimeline, 73
 canvas
 RightClickMenu, 155
 CanvasGroup
 DisplayAsFrameOnTimeline, 59
 ChangeColorOnDwell, 41
 _renderer, 43
 OnRaycastEnter, 42
 OnRaycastExit, 42
 RaycastEnterColor, 43
 RaycastExitColor, 43
 Start, 43
 ChangeTimelineScale
 Timeline, 167
ClickableOnTimeline
 ClickableOnTimeline, 43
 _rect, 48
 IsClickedAndMouseStillDown, 48
 IsHovered, 48
 OnPointerDown, 45
 OnPointerEnter, 46
 OnPointerExit, 46

OnPointerUp, 47
 Start, 47
 Timeline, 48
 Update, 47
CloseLogPanel
 LogMessagesManager, 125
Color
 VisualAngleTester, 193
Colorable, 49
 _color, 53
 _renderer, 53
 ACTIVE, 50
 AoiColors, 50
 INACTIVE, 50
 INVISIBLE, 50
 OnActivate, 51
 OnDeactivate, 51
 SetColor, 51, 52
 Start, 52
CompleteManualResizingDueToOverlap
 AOIManager, 29
ContentContainer
 LogMessagesManager, 126
CreateKeyframe
 Animation, 13
CSVtoDataSampleList
 DebugReRun, 55
CurrentEyeData
 EyeRecorder, 95
CurrentFrame
 VideoManager, 189
CurrentFrameLabel
 Playhead, 132
CurrentFrameLabelWithBG
 Playhead, 132
CurrentlyHitObject
 GazeRaycaster, 105
CurrentVRStatus
 VRManager, 205
datalist
 SaveAndLoadAOIs, 158
DebugFixationSampleFlagsToCsv
 DebugReRun, 55
DebugReRun, 54
 CSVtoDataSampleList, 55
 DebugFixationSampleFlagsToCsv, 55
 OnReRun, 55
DeleteKeyframeFromRightClickMenu
 Animation, 13
DeleteObjectUsingRightClickMenu
 RightClickMenu, 152
DetectDwells
 EventDetection, 81
DetectFixations
 EventDetection, 81
direction
 GazeToTextureMapper, 114
DisplayAsFrameOnTimeline, 56
CanvasGroup, 59
Frame, 59
 Hide, 58
 IsDraggable, 59
 RecalculateFrameBasedOnPosition, 58
Rect, 59
Show, 58
Start, 58
Timeline, 60
Update, 59
distance
 VisualAngleTester, 193
DistanceToCamera
 DraggableAroundCamera, 67
DraggableAlongXYPlane, 60
 _offset, 63
 FlipMesh, 63
 IsTryingToMove, 63
 Move, 61
 MovementCompleted, 63
 MovementStarted, 63
 OnDisable, 61
 OnMouseDown, 62
 OnMouseUp, 62
 Update, 62
DraggableAroundCamera, 63
 _offset, 67
DistanceToCamera, 67
 IsTryingToMove, 68
 IsTryingToMoveDueToBeingActivted, 68
 Move, 65
 MovementCompleted, 67
 MovementStarted, 67
 OnActivate, 65
 OnDeactivate, 66
 OnMouseDown, 66
 OnMouseUp, 66
 Update, 67
DraggableOnTimeline, 68
 _mouseOffset, 73
Canvas, 73
 IsBeingDragged, 73
 MovementCompleted, 70
 MovementCompletedSubscribers, 73
 MovementCondition, 70
 OnMovementCompleted, 71
 OnPointerDown, 71
 OnPositionChange, 71
 SetAnchoredX, 72
 Start, 72
 Update, 72
Duration
 EventDetection.Event, 78
DurationOfFirstFixation
 EventDetection.AoiParameters, 37
Dwell
 EventDetection.Event, 77
DwellCount

EventDetection.AoiParameters, 37
Dwells
 EventDetection, 86
DwellToleranceInMs
 EventDetection, 85
DwellToleranceInput
 VrSettingsMenu, 210

EditorCamera, 73
 Camera, 76
 EditorCamera_GameObject, 76
 OnRecordingToggled, 74
 OnTestSceneToggled, 75
 OnVrStatusChange, 75
 Start, 76
 Update, 76
EditorCamera_GameObject
 EditorCamera, 76
EndFrame
 Timeline, 179
EndFrameLabel
 Timeline, 178
EndTimestamp
 EventDetection.Event, 78
EntryTime
 EventDetection.AoiParameters, 37
Error
 VRManager, 202
Event
 EventDetection.Event, 77
EventCountProportion
 EventDetection, 82
EventDetection, 78
 DetectDwells, 81
 DetectFixations, 81
 Dwells, 86
 DwellToleranceInMs, 85
 EventCountProportion, 82
 EventDetection, 80
 EventDurationProportion, 83
 FilteredDataSamples, 85
 FilterInvalidData, 83
 FindIndexOfNextValidSample, 84
 FixationDispersionThresholdInDegrees, 85
 FixationDurationThresholdInMs, 85
 Fixations, 86
 Fixationsample_flags, 86
 FixationTimeBetweenSamplesLimitInMs, 86
 FixationVelocityThresholdInDegPerSecond, 86
 GetSampleIndexAtMinFixationDuration, 84
 RawDataSamples, 86
 Results, 86
 UniqueAoiNames, 85
EventDetection.AoiParameters, 36
 AoiName, 37
 AoiParameters, 36
 DurationOfFirstFixation, 37
 DwellCount, 37
 EntryTime, 37
 FirstPassDwellTime, 37
 FixationCount, 37
 ProportionOfFixations, 37
 ProportionOfTotalDwellTime, 37
 TimeToFirstFixation, 37
 TotalDwellTime, 37
EventDetection.Event, 77
 AoiName, 78
 Duration, 78
 Dwell, 77
 EndTimestamp, 78
 Event, 77
 EventType, 77
 Fixation, 77
 StartTimestamp, 78
 Type, 78
EventDetectionTester, 87
 RunTest, 88
 Start, 88
EventDetectionTester.TesterSample, 162
 aoi_test_selector, 164
 fake_time, 164
 ms_per_videoframe, 164
 TesterSample, 164
 v1, 164
 v2, 164
 v3, 164
EventDurationProportion
 EventDetection, 83
EventType
 EventDetection.Event, 77
EyeCallback
 EyeRecorder, 90
EyeFramework
 VrSettingsMenu, 210
EyeRecorder, 88
 _eye_callback_registered, 94
 _recordedEyeData, 94
 _timestampOffset, 95
 CurrentEyeData, 95
 EyeCallback, 90
 FinishRecording, 91
 Instance, 95
 IsRecording, 95
 OnApplicationQuit, 91
 OnDisable, 91
 OnRecordingToggled, 95
 PreviousEyeData, 95
 RawData, 95
 Recording_ID, 95
 Release, 92
 Sensitivity, 95
 SetIsRecordingAndNotify, 92
 Start, 93
 StartRecording, 93
 ToggleRecording, 93
 Update, 94
EyeTrackingDataSample, 96

AoiName, 97
 EyeTrackingDataSample, 97
 GazeDirection, 97
 Timestamp, 97
 VerboseEyeData, 97
 VideoFrame, 97

 fake_data
 GazeSimulator, 111
 fake_time
 EventDetectionTester.TesterSample, 164
 fake_time_in_ms
 GazeSimulator, 111
 FilteredDataSamples
 EventDetection, 85
 FilterInvalidData
 EventDetection, 83
 FindIndexOfNextValidSample
 EventDetection, 84
 FinishRecording
 EyeRecorder, 91
 FirstPassDwellTime
 EventDetection.AoiParameters, 37
 Fixation
 EventDetection.Event, 77
 FixationCount
 EventDetection.AoiParameters, 37
 FixationDispersionThresholdInDegrees
 EventDetection, 85
 FixationDispersionThresholdInput
 VrSettingsMenu, 210
 FixationDurationThresholdInMs
 EventDetection, 85
 FixationDurationThresholdInput
 VrSettingsMenu, 211
 Fixations
 EventDetection, 86
 Fixationsample_flags
 EventDetection, 86
 FixationTimeBetweenSamplesLimitInMs
 EventDetection, 86
 FixationVelocityThresholdInDegPerSecond
 EventDetection, 86
 FixationVelocityThresholdInput
 VrSettingsMenu, 211
 FixPlayheadOvershoot
 Timeline, 168
 FlipMesh
 AOIManager, 35
 DraggableAlongXYPlane, 63
 FPSCounter, 98
 _averageCounter, 99
 _averageFromAmount, 99
 _cacheNumbersAmount, 99
 _currentAveraged, 99
 _frameRateSamples, 100
 Awake, 99
 CachedNumberStrings, 100
 Text, 100

 Frame
 Update, 99
 Frame
 DisplayAsFrameOnTimeline, 59
 FrameCount
 VideoManager, 189

 GazeDirection
 EyeTrackingDataSample, 97
 GazeRaycast
 GazeRaycaster, 102
 GazeRaycaster, 101
 Awake, 102
 CurrentlyHitObject, 105
 GazeRaycast, 102
 Instance, 104
 OnNothingHit, 103
 OnObjectHit, 103
 ToggleGazeRaycastingVrStatusChange, 104
 GazeRayRenderer, 105
 _GazeRayRenderer, 108
 _LengthOfRay, 108
 _gazeSphere, 108
 _gazeSphereRenderer, 108
 _ray_color, 108
 Awake, 106
 Instance, 108
 RenderCurrentGazeAsRay, 106
 ToggleGazeRenderingOnVrStatusChange, 107
 GazeSimulator, 109
 fake_data, 111
 fake_time_in_ms, 111
 GenerateFixation, 110
 GetNextFakeSample, 111
 OnDrawGizmos, 111
 Start, 111
 TestRayDirection, 112
 UseSimulatedDataInstead, 112
 GazeToTextureMapper, 112
 direction, 114
 OnDrawGizmos, 113
 texelX, 114
 texelY, 114
 VectorToUV, 113
 VideoHeight, 114
 VideoWidth, 114
 GazeVisibilityToggle
 VrSettingsMenu, 211
 GenerateFixation
 GazeSimulator, 110
 GetChildGameObject
 RightClickMenu, 152
 GetKeyframeIndexByFrame
 Animation, 14
 GetNextFakeSample
 GazeSimulator, 111
 GetSampleIndexAtMinFixationDuration
 EventDetection, 84
 GotClicked
 RightClickMenu, 152

HandleForResizable, 114
OnMouseDown, 116
OnMouseUp, 116
Parent, 116
Start, 116
HandlePrefab
 Resizable, 148
Height
 Timeline, 179
Helper, 117
 MostCommon< T >, 117
Hide
 DisplayAsFrameOnTimeline, 58
HideHandles
 Resizable, 143
HideKeyframes
 Animation, 14
HideMenu
 RightClickMenu, 153
HoverHandoverWithChildren
 Timeline, 168

INACTIVE
 Colorable, 50
Inactive
 VRManager, 202
Instance
 AOIManager, 35
 EyeRecorder, 95
 GazeRaycaster, 104
 GazeRayRenderer, 108
 LogMessagesManager, 127
 Timeline, 179
 VideoManager, 189
InterpolatedFrames
 Animation, 23
INVISIBLE
 Colorable, 50
IsBeingDragged
 DraggableOnTimeline, 73
IsClickedAndMouseStillDown
 ClickableOnTimeline, 48
IsCurrentlyHitByRaycast
 RaycastHitHandler, 137
IsCursorOverUI
 RightClickMenu, 153
IsDraggable
 DisplayAsFrameOnTimeline, 59
IsFrameInKeyframes
 Animation, 15
IsGazeVisibleInVr
 VrSettingsMenu, 211
IsHovered
 ClickableOnTimeline, 48
 Timeline, 179
IsInitialized
 VideoManager, 189
IsObjectInCorrectState
 Animation, 15

IsPreVideoTestSceneEnabled
 VrSettingsMenu, 211
IsRecording
 EyeRecorder, 95
IsTryingToMove
 DraggableAlongXYPlane, 63
 DraggableAroundCamera, 68
IsTryingToMoveDueToBeingActivted
 DraggableAroundCamera, 68
IsTryingToResize
 Resizable, 149
IsVRCameraTracked
 VrSettingsMenu, 211
ItemPrefab
 Listable, 123

KeyframePrefab
 Animation, 23
Keyframes
 Animation, 23
 SaveAndLoadAOIs.AOISaveData, 38

LastClickedObjectWithMenu
 RightClickMenu, 155
LaunchCalibration
 VrSettingsMenu, 208
list
 SaveAndLoadAOIs.Datalist, 54
Listable, 117
 _contentContainer, 122
 _inputField, 122
 _itemInstance, 122
 ItemPrefab, 123
 ListableCount, 123
 Name, 123
 OnActivate, 120
 OnDeactivate, 120
 OnDestroy, 120
 OnListItemClick, 120
 operator string, 121
 SetName, 121
 Start, 121
 UpdateNameFromInputField, 122
ListableCount
 Listable, 123
LoadAoIs
 SaveAndLoadAOIs, 157
Loading
 VRManager, 202
LoadKeyframes
 Animation, 16
LogEntryPrefab
 LogMessagesManager, 126
LogMessagesManager, 123
 Awake, 125
 CloseLogPanel, 125
 ContentContainer, 126
 Instance, 127
 LogEntryPrefab, 126

LogPanel, 126
 OnLogMessageReceived, 125
 OpenLogPanel, 126
 ScrollRect, 126
 Update, 126
LogPanel
 LogMessagesManager, 126
Menu
 RightClickMenu, 156
menu
 VrSettingsMenu, 211
MenuCloseLogic, 127
 Update, 128
Mesh
 Resizable, 149
MostCommon< T >
 Helper, 117
Move
 DraggableAlongXYPlane, 61
 DraggableAroundCamera, 65
MovementCompleted
 DraggableAlongXYPlane, 63
 DraggableAroundCamera, 67
 DraggableOnTimeline, 70
MovementCompletedSubscribers
 DraggableOnTimeline, 73
MovementCondition
 DraggableOnTimeline, 70
 Playhead, 130
MovementStarted
 DraggableAlongXYPlane, 63
 DraggableAroundCamera, 67
ms_per_videoframe
 EventDetectionTester.TesterSample, 164
Name
 Listable, 123
 SaveAndLoadAOIs.AOISaveData, 38
NewActiveAOI
 AOIManager, 36
ObjectChangeStarted
 Animation, 16
OnActivate
 Animation, 17
 AOIComponent, 25
 Colorable, 51
 DraggableAroundCamera, 65
 Listable, 120
 Resizable, 143
OnAOIVisibilityToggled
 VRCamera, 198
 VrSettingsMenu, 211
OnApplicationQuit
 EyeRecorder, 91
OnDeactivate
 Animation, 17
 AOIComponent, 26
 Colorable, 51
 DraggableAroundCamera, 66
 Listable, 120
 Resizable, 144
OnDestroy
 Animation, 17
 AOIComponent, 26
 Listable, 120
 RightClickMenu, 154
OnDisable
 DraggableAlongXYPlane, 61
 EyeRecorder, 91
OnDrawGizmos
 GazeSimulator, 111
 GazeToTextureMapper, 113
 Resizable, 144
OnFrameCountChange
 VideoManager, 189
OnIsPreVideoTestSceneEnabledToggled
 VrSettingsMenu, 211
OnListItemClick
 Listable, 120
OnLoadAOIButtonDown
 SaveAndLoadAOIs, 158
OnLogMessageReceived
 LogMessagesManager, 125
OnMouseDown
 DraggableAlongXYPlane, 62
 DraggableAroundCamera, 66
 HandleForResizable, 116
OnMouseUp
 DraggableAlongXYPlane, 62
 DraggableAroundCamera, 66
 HandleForResizable, 116
OnMovementCompleted
 DraggableOnTimeline, 71
 Playhead, 130
 VisualAngleTester, 192
OnNewDwellToleranceEntered
 VrSettingsMenu, 208
OnNewFixationDispersionThresholdEntered
 VrSettingsMenu, 208
OnNewFixationDurationThresholdEntered
 VrSettingsMenu, 208
OnNewFixationVelocityThresholdEntered
 VrSettingsMenu, 208
OnNewFrame
 VideoManager, 189
OnNewFrameInTimeline
 VideoManager, 182
OnNewFrameInVideoPlayer
 Timeline, 169
OnNewFrameReady
 VideoManager, 183
OnNothingHit
 GazeRaycaster, 103
OnObjectHit
 GazeRaycaster, 103

OnPointerDown
 ClickableOnTimeline, 45
 DraggableOnTimeline, 71

OnPointerEnter
 ClickableOnTimeline, 46
 Timeline, 170

OnPointerExit
 ClickableOnTimeline, 46
 Timeline, 170

OnPointerUp
 ClickableOnTimeline, 47

OnPositionChange
 DraggableOnTimeline, 71
 Playhead, 131

OnPreVideoStimulusToggled
 VRCamera, 199

OnPreVideoTestSceneToggled
 PreVideoTestScene, 134

OnRaycastEnter
 ChangeColorOnDwell, 42
 RaycastHitHandler, 137

OnRaycastExit
 ChangeColorOnDwell, 42
 RaycastHitHandler, 137

OnRecordingToggled
 EditorCamera, 74
 EyeRecorder, 95
 PreVideoTestScene, 134
 Timeline, 170
 VideoManager, 184
 VRCamera, 199

OnRectTransformDimensionsChange
 Timeline, 171

OnReRun
 DebugReRun, 55

OnTargetFrameChange
 Timeline, 178

OnTestSceneToggled
 AOIManager, 30
 EditorCamera, 75

OnVideoLoaded
 VideoManager, 184

OnVideoManagerInitialized
 Timeline, 171
 VideoManager, 190

OnVideoPlayerError
 VideoManager, 185

OnVideoPlayerNewFrameCount
 Timeline, 172

OnVideoPlayerPrepareCompleted
 VideoManager, 185

OnVolumeSliderValueChanged
 VideoManager, 186

OnVRStatusChange
 VRCamera, 200

OnVrStatusChange
 AOIManager, 30
 EditorCamera, 75

 PreVideoTestScene, 135

OnXRStatusUpdate
 VRManager, 205

OpenLogPanel
 LogMessagesManager, 126

operator string
 Listable, 121

Parent
 HandleForResizable, 116

phi
 VisualAngleTester, 193

Playhead, 128
 CurrentFrameLabel, 132
 CurrentFrameLabelWithBG, 132
 MovementCondition, 130
 OnMovementCompleted, 130
 OnPositionChange, 131
 SetAnchoredX, 131
 Update, 132

PlayheadRect
 Timeline, 179

PlayPause
 VideoManager, 186

PlayUntilFrameOne
 VideoManager, 186

Position
 Animation.FrameData, 100

PreVideoTestScene, 133
 Awake, 134
 OnPreVideoTestSceneToggled, 134
 OnRecordingToggled, 134
 OnVrStatusChange, 135

PreVideoTestSceneToggle
 VrSettingsMenu, 211

PreviousEyeData
 EyeRecorder, 95

ProportionOfFixations
 EventDetection.AoiParameters, 37

ProportionOfTotalDwellTime
 EventDetection.AoiParameters, 37

RawData
 EyeRecorder, 95

RawDataSamples
 EventDetection, 86

RaycastEnterColor
 ChangeColorOnDwell, 43

RaycastExitColor
 ChangeColorOnDwell, 43

RaycastHitHandler, 136
 IsCurrentlyHitByRaycast, 137
 OnRaycastEnter, 137
 OnRaycastExit, 137

RecalculateFrameBasedOnPosition
 DisplayAsFrameOnTimeline, 58

RecenterOrigin
 Resizable, 144

RecenterOriginBasedOnKeyframeData

Resizable, 145
 RecordButton, 138
 _offSprite, 140
 _onSprite, 140
 _recButton, 140
 _recButtonImage, 140
 SetImageOnRecordingToggle, 139
 SetInteractivityBasedOnVrStatus, 139
 Start, 140
 Recording_ID
 EyeRecorder, 95
 Rect
 DisplayAsFrameOnTimeline, 59
 Release
 EyeRecorder, 92
 RemoveItem
 RightClickMenu, 154
 RenderCurrentGazeAsRay
 GazeRayRenderer, 106
 Resizable, 141
 .collider, 148
 .handles, 148
 .renderer, 148
 .vertices, 148
 HandlePrefab, 148
 HideHandles, 143
 IsTryingToResize, 149
 Mesh, 149
 OnActivate, 143
 OnDeactivate, 144
 OnDrawGizmos, 144
 RecenterOrigin, 144
 RecenterOriginBasedOnKeyframeData, 145
 Resize, 146
 ResizingCompleted, 148
 ResizingStarted, 148
 SetIsTryingToResize, 146
 ShowHandles, 147
 Start, 147
 Update, 147
 Resize
 Resizable, 146
 ResizingCompleted
 Resizable, 148
 ResizingStarted
 Resizable, 148
 Results
 EventDetection, 86
 RightClickMenu, 149
 .contentContainer, 155
 .itemPrefab, 155
 .menuPrefab, 155
 AddDeleteOnStart, 155
 .AddItem, 151
 Awake, 151
 canvas, 155
 DeleteObjectUsingRightClickMenu, 152
 GetChildGameObject, 152
 GotClicked, 152
 HideMenu, 153
 IsCursorOverUI, 153
 LastClickedObjectWithMenu, 155
 Menu, 156
 OnDestroy, 154
 RemoveItem, 154
 ShowMenu, 154
 Update, 154
 Rotation
 Animation.FrameData, 100
 RunTest
 EventDetectionTester, 88
 Save
 SaveAndLoadAOIs, 158
 SaveAndLoadAOIs, 156
 datalist, 158
 LoadAois, 157
 OnLoadAOIButtonDown, 158
 Save, 158
 SaveAndLoadAOIs.AOISaveData, 38
 Keyframes, 38
 Name, 38
 SaveAndLoadAOIs.Datalist, 53
 list, 54
 ScrollRect
 LogMessagesManager, 126
 ScrollScale
 Timeline, 179
 SeekCompleted
 VideoManager, 187
 SeekFrameInVideo
 VideoManager, 187
 Sensitivity
 EyeRecorder, 95
 SequenceChartPythonProcess, 159
 CallSequenceChartScript, 159
 SetActiveObject
 AOIManager, 31
 SetAnchoredX
 DraggableOnTimeline, 72
 Playhead, 131
 SetBlockEdgeSnapping
 Timeline, 173
 SetColor
 Colorable, 51, 52
 SetCurrentVRStatusAndNotify
 VRManager, 202
 SetEndFrame
 Timeline, 174
 SetFramerateOnVREnable
 Setup, 161
 SetImageOnRecordingToggle
 RecordButton, 139
 SetInteractivityBasedOnVrStatus
 RecordButton, 139
 SetIsRecordingAndNotify
 EyeRecorder, 92

SetIsTryingToResize
 Resizable, 146
SetKeyframeVisibility
 Animation, 18
SetName
 Listable, 121
SetObjectsStateToFrameData
 Animation, 18
SetStartFrame
 Timeline, 174
SetTargetFrame
 Timeline, 174, 175
Setup, 160
 SetFramerateOnVREnable, 161
 Start, 161
SetupRightClickMenu
 Animation, 19
SetVideoStartFrame
 Timeline, 176
Show
 DisplayAsFrameOnTimeline, 58
ShowHandles
 Resizable, 147
ShowKeyframes
 Animation, 20
ShowMenu
 RightClickMenu, 154
size
 VisualAngleTester, 193
SkyboxMaterial
 VideoManager, 189
SpawnAoi
 AOIManager, 32
SpawnAOlatCursorPosition
 AOIManager, 32
SpawnDistance
 AOIManager, 36
Speed
 CameraDrag, 40
Start
 Animation, 20
 AOIComponent, 26
 AOIManager, 33
 CameraDrag, 39
 ChangeColorOnDwell, 43
 ClickableOnTimeline, 47
 Colorable, 52
 DisplayAsFrameOnTimeline, 58
 DraggableOnTimeline, 72
 EditorCamera, 76
 EventDetectionTester, 88
 EyeRecorder, 93
 GazeSimulator, 111
 HandleForResizable, 116
 Listable, 121
 RecordButton, 140
 Resizable, 147
 Setup, 161
Timeline, 177
VideoManager, 188
VisualAngleTester, 192
VRButton, 195
VrSettingsMenu, 209
StartFrame
 Timeline, 179
StartFrameLabel
 Timeline, 179
StartManualResizingDueToOverlap
 AOIManager, 33
StartRecording
 EyeRecorder, 93
StartTimestamp
 EventDetection.Event, 78
StartXR
 VRManager, 203
StartXRCoroutine
 VRManager, 203
StopXR
 VRManager, 204
SyncTimelineWithVideo
 Timeline, 177
TargetFrame
 Timeline, 179
TargetTexture
 VideoManager, 189
TestDirection
 VisualAngleTester, 193
TesterSample
 EventDetectionTester.TesterSample, 164
TestRayDirection
 GazeSimulator, 112
texelX
 GazeToTextureMapper, 114
texelY
 GazeToTextureMapper, 114
Text
 FPSCounter, 100
Timeline, 164
 playhead, 178
 Awake, 167
 BlockedEdgeSnapping, 179
 ChangeTimelineScale, 167
 ClickableOnTimeline, 48
 DisplayAsFrameOnTimeline, 60
 EndFrame, 179
 EndFrameLabel, 178
 FixPlayheadOvershoot, 168
 Height, 179
 HoverHandoverWithChildren, 168
 Instance, 179
 IsHovered, 179
 OnNewFrameInVideoPlayer, 169
 OnPointerEnter, 170
 OnPointerExit, 170
 OnRecordingToggled, 170
 OnRectTransformDimensionsChange, 171

OnTargetFrameChange, 178
 OnVideoManagerInitialized, 171
 OnVideoPlayerNewFrameCount, 172
 PlayheadRect, 179
 ScrollScale, 179
 SetBlockEdgeSnapping, 173
 SetEndFrame, 174
 SetStartFrame, 174
 SetTargetFrame, 174, 175
 SetVideoStartFrame, 176
 Start, 177
 StartFrame, 179
 StartFrameLabel, 179
 SyncTimelineWithVideo, 177
 TargetFrame, 179
 Update, 178
 VideoStartFrame, 180
 Width, 180
 WidthPerFrame, 180
 ZoomFactor, 180
 Timestamp
 EyeTrackingDataSample, 97
 TimeToFirstFixation
 EventDetection.AoiParameters, 37
 ToggleAOIColliders
 AOIManager, 34
 ToggleAOIVisibilityInVr
 VrSettingsMenu, 209
 ToggleGazeRaycastingVrStatusChange
 GazeRaycaster, 104
 ToggleGazeRenderingOnVrStatusChange
 GazeRayRenderer, 107
 ToggleGazeVisibilityInVr
 VrSettingsMenu, 209
 ToggleKeyframeVisibilityFromRightClickMenu
 Animation, 20
 TogglePreVideoTestScene
 VrSettingsMenu, 209
 ToggleRecording
 EyeRecorder, 93
 ToggleVisibility
 VrSettingsMenu, 209
 ToggleVRCameraTracking
 VrSettingsMenu, 209
 ToggleXR
 VRManager, 204
 TotalDwellTime
 EventDetection.AoiParameters, 37
 Type
 EventDetection.Event, 78
 Ulrepresentation
 Animation.FrameData, 100
 UniqueAoiNames
 EventDetection, 85
 Update
 Animation, 21
 AOIManager, 34
 CameraDrag, 39
 ClickableOnTimeline, 47
 DisplayAsFrameOnTimeline, 59
 DraggableAlongXYPlane, 62
 DraggableAroundCamera, 67
 DraggableOnTimeline, 72
 EditorCamera, 76
 EyeRecorder, 94
 FPSCounter, 99
 LogMessagesManager, 126
 MenuCloseLogic, 128
 Playhead, 132
 Resizable, 147
 RightClickMenu, 154
 Timeline, 178
 VideoManager, 188
 VisualAngleTester, 192
 VrSettingsMenu, 210
 UpdateImageBasedOnVrManagerStatus
 VRButton, 195
 UpdateInteractivityBasedOnRecordingStatus
 VRButton, 195
 UpdateKeyframe
 Animation, 21
 UpdateNameFromInputField
 Listable, 122
 UpdateOrCreateKeyframeOnObjectChangeCompleted
 Animation, 22
 UseSimulatedDataInstead
 GazeSimulator, 112
 v1
 EventDetectionTester.TesterSample, 164
 v2
 EventDetectionTester.TesterSample, 164
 v3
 EventDetectionTester.TesterSample, 164
 VectorToUV
 GazeToTextureMapper, 113
 VerboseEyeData
 EyeTrackingDataSample, 97
 Vertices
 Animation.FrameData, 100
 VideoFrame
 Animation.FrameData, 101
 EyeTrackingDataSample, 97
 VideoHeight
 GazeToTextureMapper, 114
 VideoManager, 180
 _isSeeking, 189
 Awake, 182
 CurrentFrame, 189
 FrameCount, 189
 Instance, 189
 IsInitialized, 189
 OnFrameCountChange, 189
 OnNewFrame, 189
 OnNewFrameInTimeline, 182
 OnNewFrameReady, 183
 OnRecordingToggled, 184

OnVideoLoaded, 184
OnVideoManagerInitialized, 190
OnVideoPlayerError, 185
OnVideoPlayerPrepareCompleted, 185
OnVolumeSliderValueChanged, 186
PlayPause, 186
PlayUntilFrameOne, 186
SeekCompleted, 187
SeekFrameInVideo, 187
SkyboxMaterial, 189
Start, 188
TargetTexture, 189
Update, 188
VideoPlayer, 190
VolumeSlider, 189

VideoPlayer
 VideoManager, 190

VideoStartFrame
 Timeline, 180

VideoWidth
 GazeToTextureMapper, 114

Visibility
 Animation.FrameData, 101

VisualAngleTester, 190
 _angle_sphere, 192
 _meshRenderer, 192
 _movementComponent, 192
 Color, 193
 distance, 193
 OnMovementCompleted, 192
 phi, 193
 size, 193
 Start, 192
 TestDirection, 193
 Update, 192

VolumeSlider
 VideoManager, 189

VRButton, 193
 _errorSprite, 196
 _loadingSprite, 196
 _offSprite, 196
 _onSprite, 196
 _vrButton, 196
 _vrButtonImage, 196
 Start, 195
 UpdateImageBasedOnVrManagerStatus, 195
 UpdateInteractivityBasedOnRecordingStatus, 195

VRCamera, 197
 Awake, 198
 OnAOIVisibilityToggled, 198
 OnPreVideoStimulusToggled, 199
 OnRecordingToggled, 199
 OnVRStatusChange, 200

VRManager, 200
 _EyeFramework, 205
 _XRRig, 205
 Active, 202

CurrentVRStatus, 205
Error, 202
Inactive, 202
Loading, 202
OnXRStatusUpdate, 205
SetCurrentVRStatusAndNotify, 202
StartXR, 203
StartXRCoroutine, 203
StopXR, 204
ToggleXR, 204
VRStatus, 202

VrSettingsMenu, 206
 AOIVisibilityToggle, 210
 AreAOIsVisibleInVr, 210
 CameraTrackingToggle, 210
 DwellToleranceInput, 210
 EyeFramework, 210
 FixationDispersionThresholdInput, 210
 FixationDurationThresholdInput, 211
 FixationVelocityThresholdInput, 211
 GazeVisibilityToggle, 211
 IsGazeVisibleInVr, 211
 IsPreVideoTestSceneEnabled, 211
 IsVRCameraTracked, 211
 LaunchCalibration, 208
 menu, 211
 OnAOIVisibilityToggled, 211
 OnIsPreVideoTestSceneEnabledToggled, 211
 OnNewDwellToleranceEntered, 208
 OnNewFixationDispersionThresholdEntered, 208
 OnNewFixationDurationThresholdEntered, 208
 OnNewFixationVelocityThresholdEntered, 208
 PreVideoTestSceneToggle, 211
 Start, 209
 ToggleAOIVisibilityInVr, 209
 ToggleGazeVisibilityInVr, 209
 TogglePreVideoTestScene, 209
 ToggleVisibility, 209
 ToggleVRCameraTracking, 209
 Update, 210

VRStatus
 VRManager, 202

Width
 Timeline, 180

WidthPerFrame
 Timeline, 180

ZoomFactor
 Timeline, 180