



Dokumentation v1.3

Inhalt

Über dieses Dokument	3
Über Retronator	3
Retro Style Erstellung	3
Einfache Verwendung	3
Hohe Leistung	3
Funktionsreichtum	3
Retronator in das Projekt einbinden	3
Das Inspector-Fenster	4
Settings	4
How to access at runtime	4
Resolution Mode	4
Blur the Image	5
Screen Tint Color	6
Colors	6
Greyscale?	6
Additional Game Cameras	6
Fading Test	7
Fade Duration	7
Fade relative	7
Blur the Fading	7
Fade Buttons	7
Zugriff per Code	8
Die Klasse benutzen	8
Methoden für Einstellungen	8
SetPixeliciousness	8
GetPixeliciousness	8
SetFixedResolution	8
GetFixedResolution	8
SetFixedWidth	8

GetFixedWidth.....	8
SetFixedHeight.....	8
GetFixedHeight.....	8
SetFixedMax	9
GetFixedMax	9
SetMode	9
GetMode	9
SetTintColor.....	9
GetTintColor	9
SetBlur	9
GetBlur.....	9
SetGreyscale.....	9
GetGreyscale	9
SetColorDepth	9
GetColorDepth	9
AddGameCamera	10
RemoveGameCamera.....	10
Methoden für Fades	10
FadeFromMinPixels	10
FadeFromNative	10
FadeToMinPixels	10
FadeToNative.....	10
Kompatibilität mit anderen Plugins.....	10
Allgemein.....	10
UFPS (Ultimate FPS).....	11
Fehlerbehebung	12

Über dieses Dokument

Dies ist die offizielle Dokumentation des Unity Assets “Retronator”, einem Script, dass es dir erlaubt, dein Spiel auf einfache und effiziente Weise zu verpixeln und zu kolorieren!

- Falls du dieses Asset nicht im Unity Asset Store oder direkt vom Autor erworben hast, handelt es sich um eine illegal Kopie und es ist dir rechtlich nicht gestattet, es zu benutzen. Informiere den Autor unverzüglich darüber, woher du diese Kopie bezogen hast! Kontaktinformationen sind verfügbar auf <http://www.garvin-gurbat.de>

Über Retronator

Retro Style Erstellung

Retronator ist ein einfach zu verwendender Hochleistungsfilter, mit dem die Kameraansicht in einem einstellbaren klassischen Retrolook-Stil gerendert werden kann. Verpasse deinem Spiel ein Spielgefühl wie in der guten alten Zeit! Back to the roots!

Einfache Verwendung

Retronator ist dazu designt, so einfach wie möglich zu sein. Die Einbindung findet per einfachem Drag and Drop statt und alle Einstellungen können im Inspector vorgenommen werden. Es gibt Vorschaufunktionen für Übergänge und der Zugriff zur Laufzeit ist sehr einfach gestaltet, da alle Methoden statisch sind und von jedem Script aufgerufen werden können.

Hohe Leistung

Retronator ist mit Wert auf Renderleistung entwickelt. Du wirst definitiv keinen Leistungsverlust feststellen – tatsächlich wird deine Bildrate in großen oder komplexen Szenen sogar erhöht!

Funktionsreichtum

Es gibt für jede Verwendung viele Einstellungsmöglichkeiten und Fades (Übergänge) zwischen pixeligen und nativen Auflösungen sind verfügbar.

Retronator in das Projekt einbinden

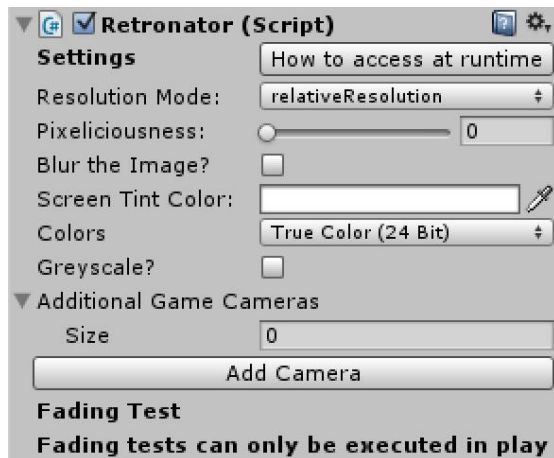
Um Retronator einzubinden, ziehe einfach das Script “**Retronator.cs**” auf deine **Kamera**.

- ✓ Das war’s! Du kannst Retronator nun benutzen.

Das Inspector-Fenster

Settings

Du findest die Einstellungen in der **Retronator-Komponente** der **Kamera**.



Hier kannst du Retronator an deine Bedürfnisse anpassen.

- Du wirst den Effekt nur in der **Game View** sehen! Seit Retronator 1.3 ist es nicht mehr nötig, in den Play Mode zu wechseln!

How to access at runtime

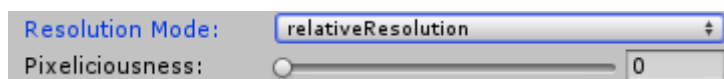
Durch einen Klick auf diesen Button wird ein Dialogfenster geöffnet, das dir ein paar Informationen zu dem Zugriff auf Retronatorfunktionen zur Laufzeit gibt. Weitere Informationen findest du im Abschnitt "Zugriff per CodeZugriff per Code".

Resolution Mode

Du hast fünf Optionen, die im Folgenden erläutert werden.

Relative Resolution Mode

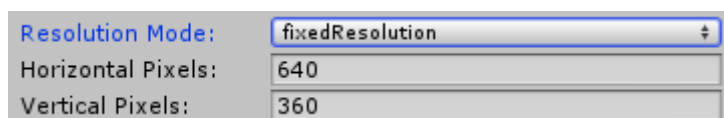
Die Einstellung "relativeResolution" erlaubt dir, die Verpixelung der Kameraansicht einfach mit einem Schieberegler oder einem Eingabefeld einzustellen.



- ✓ Jeder Fließkommawert von 0f (inklusive) bis 100f (inklusive) ist erlaubt.
0 bedeutet, dass das Bild nicht verpixelt wird. Die native Auflösung wird verwendet.
100 bedeutet maximale Verpixelung und du wirst nur rein paar Pixel auf dem Bildschirm sehen.
- Werte unter 0 oder über 100 sind nicht erlaubt!

Fixed Resolution Mode

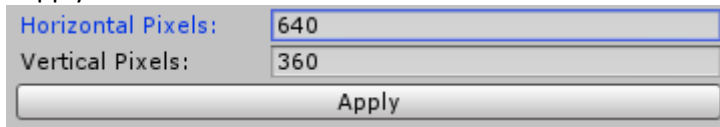
Der "fixedResolution"-Modus erlaubt dir, eine Breite und Höhe einzugeben. Deine Kameraansicht wird in exakt dieser Auflösung gerendert.



- ✓ Alle durch zwei teilbaren Werte (gerade Zahlen) sind erlaubt! Die Werte sollten nicht höher als die native Auflösung des Spiels sein und das selbe Seitenverhältnis darstellen.

- Ungerade Zahlen sind nicht erlaubt!

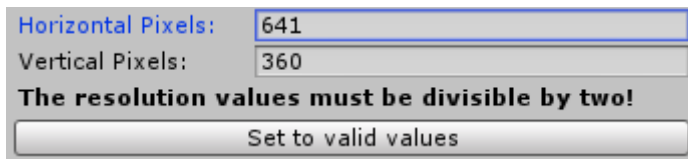
Während du Werte eingibst, werden diese überprüft. Falls sie gültig sind, kannst du sie mit dem "Apply"-Button übernehmen.



A screenshot of a dialog box with two input fields. The first field is labeled "Horizontal Pixels:" and contains the value "640". The second field is labeled "Vertical Pixels:" and contains the value "360". Below these fields is a button labeled "Apply".

- Die Werte müssen mit dem "Apply"-Button bestätigt werden. Andernfalls werden sie ignoriert!

Falls du unzulässige Werte eingibst, wirst du darüber informiert und musst sie korrigieren. Du kannst dies entweder manuell tun, indem du neue Werte eingibst, oder automatisiert durch Klicken des "Set to valid values"-Buttons erledigen lassen. Vergiss nicht, die Werte danach zu bestätigen.



A screenshot of a dialog box. The "Horizontal Pixels:" field contains the value "641". The "Vertical Pixels:" field contains the value "360". Below these fields is a bold error message: "The resolution values must be divisible by two!". At the bottom is a button labeled "Set to valid values".

Fixed Width Mode

Der "fixedWidth"-Modus erlaubt dir, einen Wert für die horizontale Auflösung einzugeben. Die vertikale Auflösung (Höhe) wird entsprechend des Seitenverhältnisses automatisch berechnet.

- Der Wert muss wie in "Fixed Resolution Mode" beschrieben zulässig sein und wie zuvor beschrieben bestätigt werden.

Fixed Height Mode

Der "fixedHeight"-Modus erlaubt dir, einen Wert für die vertikale Auflösung einzugeben. Die horizontale Auflösung (Breite) wird entsprechend des Seitenverhältnisses automatisch berechnet.

- Der Wert muss wie in "Fixed Resolution Mode" beschrieben zulässig sein und wie zuvor beschrieben bestätigt werden.

Fixed Max Mode

Der "fixedMax"-Modus erlaubt dir die Eingabe einer Auflösung für die größere Kante der Kameraansicht. Der Filter wird das aktuelle Seitenverhältnis ermitteln und die Auflösung der größeren Kante auf den eingegebenen Wert setzen. Üblicherweise bedeutet das, dass in Landscape-Auflösungen die Breite auf den gegebenen Wert eingestellt wird und in Portrait-Auflösungen die Höhe. Die Auflösung der anderen Kante (Höhe in Landscape und Breite in Portrait) wird entsprechend des Seitenverhältnisses berechnet.

- Der Wert muss wie in "Fixed Resolution Mode" beschrieben zulässig sein und wie zuvor beschrieben bestätigt werden.

Blur the Image

Diese Einstellung erlaubt das Weichzeichnen des Bildes. Das gibt dem Bild eine verschwommene Optik.

Aktiviere die Checkbox, um das Weichzeichnen zu aktivieren. Deaktiviere sie, um quadratische Pixel zu bekommen.

Screen Tint Color

Diese Option erlaubt die Auswahl einer Farbe, um das Bild einzufärben.

- Verändere den Alphakanal der Farbe nicht, da dies zu ungewünschten Effekten führen kann.

Colors

Diese Option erlaubt dir, die Anzahl der Farben einzustellen, in denen die Szene gerendert wird. Dies kann dazu beitragen, den Retro-Effekt des Spiels zu erhöhen. Mögliche Optionen sind:

True Color (24 Bit)

Das Bild wird normal aus einer Palette von 16,77 Mio. Farben berechnet.

32k Colors (15 Bit)

Die Farbpalette wird auf 32.000 Farben reduziert.

4k Colors (12 Bit)

Die Farben werden an eine Palette mit 4.000 Farben angepasst.

512 Colors (9 Bit)

Das Bild wird auf 512 Farben heruntergerechnet.

64 Colors (6 Bit)

Es werden nur 64 Farben benutzt, um das Bild darzustellen.

8 Colors (3 Bit)

Die Szene wird mit nur 8 Farben sehr minimalistisch dargestellt.

- Ist „Greyscale“ aktiviert, bestimmt „Colors“ die Anzahl der Graustufen.

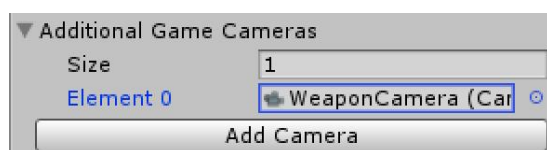
Greyscale?

Durch Aktivierung dieser Checkbox wird die Szene nur in Graustufen (schwarz/weiß) gerendert.

- Die Anzahl der möglichen Graustufen wird durch die Option „Colors“ bestimmt.

Additional Game Cameras

Solltest du weitere Spielekameras (Kameras, die die Szene und nicht die Benutzeroberfläche darstellen) benutzen, füge sie in diese Liste ein. Die Hauptkamera, auf der das Script liegt, sollte hier nicht hinzugefügt werden.

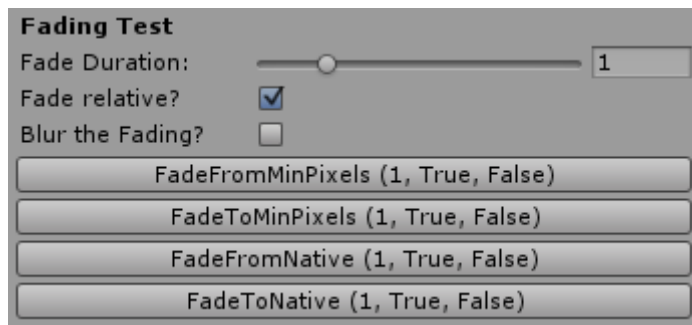


Mit dem Schalter „Add Camera“ fügst du einen weiteren Eintrag in der Liste hinzu. Solltest du die Liste nicht sehen, klicke bitte auf das Dreieck vor dem Text „Additional Game Cameras“. Du kannst eine weitere Kamera in die Liste eintragen, indem du das GameObject mit der Kamera aus der Szene in den Slot ziehst, oder auf den kleinen Kreis rechts neben einem Eintrag klickst und die Kamera auswählst.

Du kannst eine Kamera wieder aus der Liste entfernen, indem du mit der rechten Maustaste auf den Eintrag klickst und „Delete Array Element“ anklickst.

Fading Test

In diesem Bereich können Fades (Transformationen), die Retronator unterstützt, getestet werden.



- Die Fade Tests sind nur im **Play Mode** verfügbar!

Fade Duration

Stelle diesen Wert auf die Zeit in Sekunden ein, die der Fade dauern soll.

Fade relative

Falls diese Checkbox aktiviert ist, werden die Fades in Bezug auf die Einstellungen vorgenommen. Im Abschnitt "Fade Buttons" wird dies erläutert.

Blur the Fading


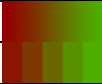
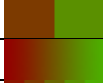
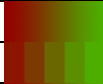

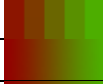



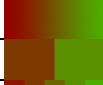
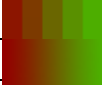
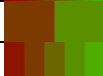



- Falls diese Checkbox aktiviert ist, wird während des Fades die Weichzeichnung aktiviert, selbst wenn "Der Wert muss wie in "Fixed Resolution Mode" beschrieben zulässig sein und wie zuvor beschrieben bestätigt werden.
- Blur the Image" nicht aktiviert ist. Nach dem Fade wird die Weichzeichnung wieder entsprechend der Einstellung von "Der Wert muss wie in "Fixed Resolution Mode" beschrieben zulässig sein und wie zuvor beschrieben bestätigt werden.


Blur the Image" deaktiviert oder weiter beibehalten.


Fade Buttons

Es gibt vier Buttons, die dir das Testen der Fades erlauben.

Die Auflösung, mit der die Fades beginnen und enden, hängt von der Einstellung von "Fade relative" ab. Die folgende Tabelle veranschaulicht die Start- und Zielauflösungen abhängig von dem Button und "Fade relative".

Button	Fade relative	Startauflösung	Zielauflösung
FadeFromMinPixels	<input type="checkbox"/>		
FadeFromMinPixels	<input checked="" type="checkbox"/>		
FadeToMinPixels	<input type="checkbox"/>		
FadeToMinPixels	<input checked="" type="checkbox"/>		
FadeFromNative	<input type="checkbox"/>		
FadeFromNative	<input checked="" type="checkbox"/>		
FadeToNative	<input type="checkbox"/>		
FadeToNative	<input checked="" type="checkbox"/>		

 = native Auflösung

 = verpixelte Auflösung entsprechend der Einstellungen



= maximale Verpixelung (nur wenige, große Pixel)

Nachdem du einen Fade getestet hast, kannst du über den “ResetFade()”-Button die Verpixelung auf die Einstellung des Retronators zurücksetzen.

Zugriff per Code

Auf die Einstellungen und Methoden von Retronator aus deinen eigenen Scripts zuzugreifen ist sehr einfach und erlaubt dir, den Effekt an deine Bedürfnisse anzupassen und jederzeit Fades zu starten.

- Zugriff per Script ist nur verfügbar, wenn das „Retronator.cs“-Script zuvor auf die Kamera gezogen wurde!

Die Klasse benutzen

Du musst alle Funktionen mit dem Klassennamen als Präfix aufrufen. Der Aufruf der Methode “SetBlur()” würde z.B. so aussehen: “Retronator.SetBlur()”. Die Methoden können in jedem Script der Szene aufgerufen werden!

Methoden für Einstellungen

SetPixeliciousness

```
void SetPixeliciousness(float rate, bool setMode = true)
```

Setzt den “Pixeliciousness”-Wert und stellt den “Resolution Mode” optional auf “relativeResolution”.

GetPixeliciousness

```
float GetPixeliciousness()
```

Liest die aktuelle eingestellte Pixeliciousness aus.

SetFixedResolution

```
void SetFixedResolution(int width, int height, bool setMode = true)
```

Setzt die “Horizontal Pixels” und “Vertical Pixels” der „fixedResolution“ und optional den “Resolution Mode” auf „fixedResolution”.

GetFixedResolution

```
Vector2 GetFixedResolution()
```

Liest die aktuelle eingestellten „Horizontal Pixels“ und „Vertical Pixels“ der „fixedResolution“ aus.

SetFixedWidth

```
void SetFixedWidth(int width, bool setMode = true)
```

Setzt die “Horizontal Pixels” der “fixedWidth” und optional den “Resolution Mode” auf „fixedWidth”.

GetFixedWidth

```
int GetFixedWidth()
```

Liest die aktuelle eingestellten „Horizontal Pixels“ der „fixedWidth“ aus.

SetFixedHeight

```
void SetFixedHeight(int height, bool setMode = true)
```

Setzt die “Vertical Pixels” der „fixedHeight“ und optional den “Resolution Mode” auf “fixedHeight”.

GetFixedHeight

```
int GetFixedHeight()
```

Liest die aktuelle eingestellten „Vertical Pixels“ der „fixedHeight“ aus.

SetFixedMax

```
void SetFixedMax(int maximum, bool setMode = true)
```

Setzt die „Maximum Pixels“ der „fixedMax“ und setzt den „Resolution Mode“ optional auf „fixedMax“.

GetFixedMax

```
int GetFixedMax()
```

Liest die aktuelle eingestellten „Maximum Pixels“ der „fixedMax“ aus.

SetMode

```
void SetMode(ResolutionMode mode)
```

Setzt den „Resolution Mode“ auf „relativeResolution“, „fixedResolution“, „fixedWidth“, „fixedHeight“ oder „fixedMax“. Du musst `Retronator.ResolutionMode.relativeResolution`, `Retronator.ResolutionMode.fixedResolution`, `Retronator.ResolutionMode.fixedWidth`, `Retronator.ResolutionMode.fixedHeight` oder `Retronator.ResolutionMode.fixedMax` für den „mode“-Parameter nutzen.

GetMode

```
ResolutionMode GetMode()
```

Liest den aktuelle eingestellten „Resolution Mode“ aus.

SetTintColor

```
void SetTintColor(Color color)
```

Setzt die „Tint Color“, die die Kameraansicht einfärbt.

GetTintColor

```
Color GetTintColor()
```

Liest die aktuell eingestellte „Tint Color“ aus.

SetBlur

```
void SetBlur(bool blur)
```

Stellt ein, ob das Bild weichgezeichnet werden soll.

GetBlur

```
bool GetBlur()
```

Liest aus, ob der Weichzeichnungsfilter („Blur the Image“) aktiviert ist.

SetGreyscale

```
void SetGreyscale(bool active)
```

Aktiviert oder deaktiviert die Darstellung in Graustufen.

GetGreyscale

```
bool GetGreyscale()
```

Liest aus, ob die Darstellung in Graustufen „Greyscale“ aktiviert ist.

SetColorDepth

```
bool SetColorDepth(int depth)
```

Setzt die Bittiefe der Farbdarstellung.

Erlaubte Werte sind: 24, 15, 12, 9, 6 und 3 (siehe [Colors](#)).

GetColorDepth

```
int GetColorDepth()
```

Liest die aktuelle Bittiefe der Farbdarstellung aus (siehe [Colors](#)).

AddGameCamera

```
bool AddGameCamera(Camera camera)
```

Fügt der Liste der “Additional Game Cameras” eine Kamera hinzu (siehe [Additional Game Cameras](#)). Gibt bei Erfolg *true* zurück. Bei *false* konnte die Kamera nicht hinzugefügt werden oder war bereits in der Liste vorhanden.

RemoveGameCamera

```
bool RemoveGameCamera(Camera camera)
```

Entfernt die angegebene Kamera aus der Liste der “Additional Game Cameras” (siehe [Additional Game Cameras](#)).

Gibt bei Erfolg *true* zurück. Bei *false* konnte die Kamera nicht entfernt werden oder war nicht in der Liste vorhanden.

Methoden für Fades

Diese Methoden werden genutzt, um eine Fade-Animation (Übergang) zu starten. Alle Parameter sind optional. Die exakte Funktionsweise der Parameter wird in “Du kannst eine Kamera wieder aus der Liste entfernen, indem du mit der rechten Maustaste auf den Eintrag klickst und „Delete Array Element“ anklickst.

Fading Test” erläutert.

“durationInSeconds” entspricht der “Fade Duration”.

“toPixeliciousness” und “fromPixeliciousness” entspricht dem “Fade relative”-Flag.

“blur” entspricht dem “Blur the Fading”-Flag.

FadeFromMinPixels

```
void FadeFromMinPixels(float durationInSeconds = 1f, bool toPixeliciousness = true, bool blur = false)
```

FadeFromNative

```
void FadeFromNative(float durationInSeconds = 1f, bool toPixeliciousness = true, bool blur = false)
```

FadeToMinPixels

```
void FadeToMinPixels(float durationInSeconds = 1f, bool fromPixeliciousness = true, bool blur = false)
```

FadeToNative

```
void FadeToNative(float durationInSeconds = 1f, bool fromPixeliciousness = true, bool blur = false)
```

Kompatibilität mit anderen Plugins

Allgemein

Grundsätzlich kann es bei der Verwendung mehrerer Plugins zu unerwünschten Resultaten kommen. Sollten Probleme bei der Benutzung von Retronator mit anderen Plugins auftreten, helfe ich dir gern weiter. Auf <http://www.garvin-gurbat.de> findest du entsprechende Kontaktinformationen.

UFPS (Ultimate FPS)

UFPS ist ein herausragendes First Person Shooter Plugin, welches mehrere Kameras benutzt. Du findest es hier im Asset Store: <https://www.assetstore.unity3d.com/en/#!/content/2943>

Da das Plugin mehrere Kameras nutzt, muss Retronator korrekt konfiguriert werden, damit es nicht zu Problemen kommt. Beispielsweise kann es passieren, dass die Waffe im Spiel nicht angezeigt wird. Um dies und weitere Fehler zu vermeiden, füge das Retronator-Script auf jedem Fall der richtigen Kamera hinzu. Dies ist die „FPSCamera“. Weiterhin ist es wichtig, dass die „WeaponCamera“ der Liste der „Additional Game Cameras“ hinzugefügt wird. Die Namen der beiden Kameras entsprechen den Namen im Beispielprojekt von UFPS. Eine genaue Erklärung zu den „Additional Game Cameras“ findest du [hier](#).

Vielen Dank an dieser Stelle an die Entwickler von UFPS für die Zusammenarbeit zwecks Kompatibilitätsmaximierung!

Fehlerbehebung

Falls du Probleme bei der Verwendung von Retronator hast, findest du die Lösung vermutlich hier. Für weitere Informationen und direkten Kontakt besuche <http://www.garvin-gurbat.de>.

Q: Warum verpixelt der Retronator mein Spiel nicht?

A: Dafür gibt es mehrere mögliche Gründe:

- Du hast das Script nicht deiner Kamera hinzugefügt
- Der Retronator ist auf keinen oder einen sehr kleinen Verpixelungseffekt eingestellt
- Der Retronator ist deaktiviert

Q: Warum wird mein User Interface nicht beeinflusst?

A: Retronator beeinflusst das UI nicht, falls dessen Canvas' Render Mode auf "Screen Space - Overlay" steht.

Falls du möchtest, dass der Retronator dein UI beeinflusst, musst du dessen Canvas' Render Mode auf "Screen Space - Camera" oder "World Space" stellen und die Kamera, die das Retronator-Script enthält als die Render Camera des Canvas' einstellen. Retronator ist nicht dazu entwickelt, UI-Elemente zu beeinflussen. Die korrekten Werte Einstellungen für eine UI zu finden, die durch Retronator verändert werden soll. Es ist aber definitiv möglich.

Q: Warum verschwinden manche Objekte, wenn ich das Spiel starte?

A: Vermutlich nutzt du mehrere Kameras zum Rendern der Szene. Du musst alle Kameras (außer der Kamera, auf der das Script liegt und GUI-Kameras) zu „Additional Game Cameras“ hinzufügen.