Eigene Spiele Präsentieren Teil 2 – GIF-Animationen

Description

Auf Twitter, Facebook und anderen Plattformen erfreuen sich GIF-Animationen großer Beliebtheit. Obwohl das Format aus der Computersteinzeit stammt, sind auch Spieleentwickler immer mehr darauf angewiesen, das eigene Spiel in GIF-Häppchen zu präsentieren.

Was ist GIF?

Bei vielen Spielen merkt man auf Screenshots kaum, das sie nur 256 Farben haben.

GIF steht für Graphics Interchange Format und ist ein uraltes Grafikformat, welches bereits 1987 von CompuServe eingeführt wurde. Die Vorteile des Format bestand in der hohen Kompression (folglich kleine Dateigrößen), der Transparenz und der Möglichkeit, Animationen zu zeigen. Diese Eigenschaft macht es heute noch so populär, da den Rest JPG und PNG mittlerweile wesentlich besser machen.

Die Kompression von GIF ist zwar im Sinne von sich bildenden Kompressionsartefakt verlustfrei (es bilden sich keine) aber das Format hat ein sehr großes Problem: Die Bilder sind auf 256 Farben beschränkt. Retro-Animationen und viele Indie-Spiele, die mit Farben sehr sparsam umgehen, leiden bei Animationen nur geringfügig darunter. Sobald man aber in ein High Color-Spiel geht, bekommt man bei Animationen große Probleme.

Bei Spielern sind diese Animationen so beliebt, weil sie in sehr kurzer Zeit viel zeigen können und meistens, wenn nichts anderes eingestellt wurde, automatisch starten. Man scrollt nur runter und sieht sofort, was einem der Entwickler mitteilen möchte.

GIF-Schwierigkeiten

So ein Pixelbrei findet keine Beachtung.

Um Probleme lösen zu können, muss man sich derer bewusst werden. Bei einem einzelnen Bild macht es in vielen Fällen nicht so viel aus, wenn man die Farben stark reduziert. Teilweise sehen diese Grafiken im Gamer-Bereich sogar besser aus. Das Problem tritt vor allem bei Animationen auf.

Je nach Auflösung können GIF-Animationen eine enorme Dateigröße erreichen. itch.io akzeptiert nur GIFs bis 3MB, je nach Spiel und Kompression kann eine zehnsekündige Animation bereits 6MB oder mehr ausmachen.

Die Größe definiert sich vorwiegend über die Bildauflösung, die Länge, die Frames pro Sekunde und die Möglichkeit, die Farbpalette zu optimieren. Wenn der Hintergrund statisch, die Bewegungen etwas langsamer sind und man nicht viele Farbwechsel hat, wird die GIF-Animation wesentlich schöner als wenn ein Einhorn einen Regenbogen auf das Spiel kotzt.

Ein weiteres, sehr zentrales Problem ist die Plattform, welche zum upload benutzt wird. Facebook und Twitter wandeln die hochgeladenen Dateien noch einmal um. Gerade bei Twitter, meiner Meinung nach dem wichtigsten sozialen Netzwerk für Spieleentwicker, artet das in einen Pixelbrei aus, wenn man es nicht richtig macht.

Bei den nachfolgenden Tipps gehe ich vom schlimmsten Szenario aus, weil es mir selbst widerfahren ist. CYPEST ist ein sehr schnelles Spiel, bei dem sich der Hintergrund zu jeder Millisekunde ändert. Es gibt viele rote Wände, Farbverläufe, Lichter und Schatten. Selbst die Darstellung der meisten Ego-Shooter gestaltet sich als GIF leichter.

Die Aufnahme

Wir nehmen natürlich keine GIF-Animation im Spiel auf sondern zunächst ein Video. Über die Tücken der Videoaufnahme habe ich bereits im ersten Teil der Serie einiges geschrieben. Obwohl das Video

später noch mindestens zwei Mal komprimiert wird (erst als GIF und dann auf Twitter erneut) müssen wir eine möglichst perfekte Ausgangssituation schaffen.

Neben der üblichen Probleme mit Codecs haben wir noch den Fall, dass vor allem Rottöne in Videos etwas blasser sind. Um eine möglichst gute Qualität zu erzielen nehmen wir Fraps, auch wenn die Dateien verhältnismäßig groß werden. Qualität hat Vorrang.

Anschließend muss die Videodatei möglichst auf den Punkt geschnitten werden, den wir zeigen wollen. Die Animation sollte nicht mehr als 15 Sekunden lang sein, besser geringer. Perfekt sind Animationen die sich als Schleife eignen, aber das ist in Spielen nur schwer zu realisieren. Für den verlustfreien Schnitt nehmen wir Avidemux. Da können wir nicht nur das Video schneiden sondern gleich die Tonspur entfernen, was uns eine kleinere Videodatei beschwert.

Umwandlung in GIF

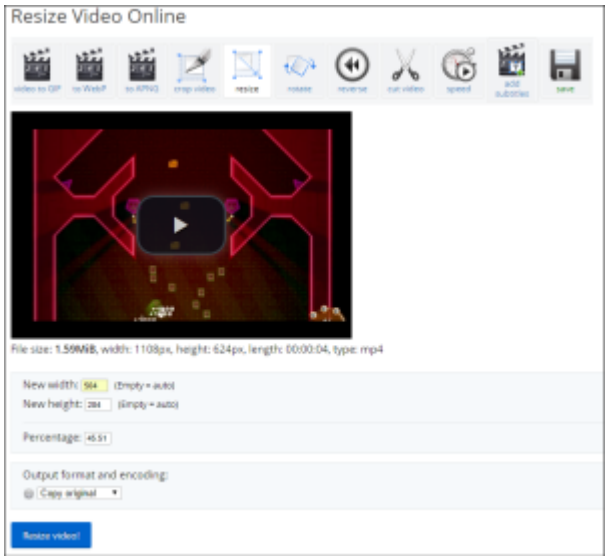
Wir schneiden das Video an den Seiten und oben ab.

Es gibt zahlreiche, teils kommerzielle Tools, mit denen man Videos in GIF umwandeln kann. Zumindest von den freien Tools habe ich dutzende ausprobiert und wieder in die Tonne getreten. Das meiner Meinung nach beste freie Werkzeug ist daher ezgif.com. Einziges Manko: Die hochgeladenen Dateien dürfen nicht größer als 120MB sein. Deshalb ist es auch so wichtig, schon am Anfang die Datei klein zu halten und sich von der unnötigen Tonspur zu trennen. Sollte die Datei immer noch zu groß sein, hilft es, die Sequenz zu kürzen oder das Spiel in einer geringeren Auflösung aufzunehmen. 720p reichen völlig aus, da die GIF noch kleiner wird.

ezgif.com unterstützt zahlreiche Videoformate, weshalb es hier kaum Probleme geben sollte. Nach dem Upload sollten wir zunächst die Bildgröße anpassen.

Nun haben wir ein weiteres Problem. Es gibt keine Auflösung, die für alle Plattformen perfekt ist. itch.io empfiehlt 630×500 und hätte das auch bei GIF-Animationen ganz gerne (minimum 315×250). Bei Facebook ist das schon komplizierter, weil hier die Größen regelmäßig geändert werden und es auch darauf ankommt, wo man das Bild / die Animation platziert. Fotos werden derzeit als 403×403 angezeigt, als Upload-Größe wird 1200×1200 empfohlen. Bei Animationen sollte in den meisten Fällen 403×227 genügen. Bei Twitter hat sich 504×284 ganz gut bewährt, da mit dieser Größe durch die Twitter-Kompression nicht so viele Verluste auftreten.

Der richtige Schnitt



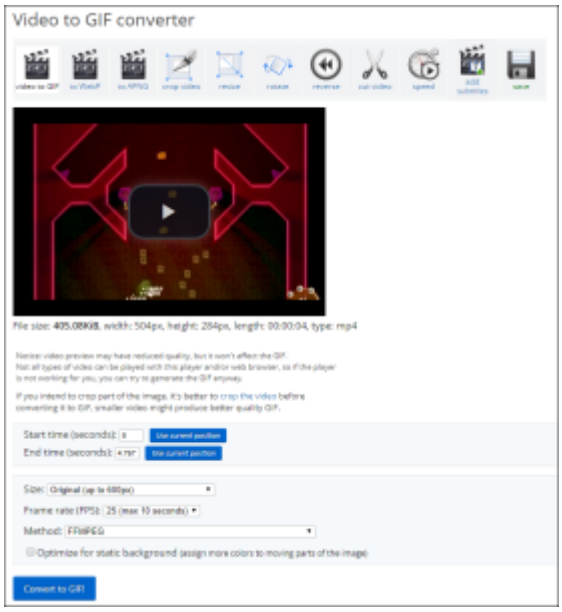
Anschließend bringen wir es auf die richtige Größe.

Neben der Größe in Pixel ist noch wichtig, ob wir das ganze Bild reduzieren oder nur einen Ausschnitt zeigen. Hier kommt es darauf an, ob wir auf die GUI verzichten können, die meistens oben, unten und am Bildschirmrand klebt. Im Zentrum steht meistens die Spielfigur und das, was tatsächlich im Spiel passiert. Wenn wir einfach nur das Bild verkleinern, können die relevanten Figuren zu klein werden. Bei CYPEST gibt es zudem noch den schwarzen Rand auf den Seiten. Mein erster Schritt besteht somit darum, die Ränder und die obere HUDs wegzulassen, indem ich es im passenden Seitenverhältnis beschneide. Das Resultat kann durchaus größer sein als unser Ziel, also 504×284. Wichtig ist das Seitenverhältnis. 788×445 würde demnach auch gehen.

Sobald wir den passenden Ausschnitt haben, reduzieren wir die Animation auf die endgültige Größe. Anschließend sehen wir, ob die Qualität und die Größe passen. Für Twitter sollte die Dateigröße möglichst unter 10MB bleiben. Ist dies nicht erreicht, kann man mit der Anzahl der Frames spielen. ezgif.com bietet zahlreiche Möglichkeiten zur Optimierung an, die meisten sind aber mit Qualitätseinbußen verbunden.

Wenn man mit den Optimierungen fertig ist, kann man sich die Animation bequem herunterladen und auf der gewünschten Plattform wieder hochladen.

Optimale Planung



Nun konvertieren wir es in GIF.

Mittlerweile denke ich, dass man den Bereich Marketing bei der Planung eines Spiels stark berücksichtigen muss. Viele denken dabei ein coole Features, eine tolle Spielwelt und ein einzigartiges Spielerlebnis, aber für das Marketing ist das zunächst Zweitrangig. Das sind meistens Daten, die irgendwo auf einer Checkliste stehen. Spieler klicken nicht jedes Spiel an um herauszufinden, ob es ein bestimmtes Feature hat. Spieler klicken, wenn sie von dem was sie sehen angezogen werden.

Da die Videokompression auf Twitter derzeit grausam ist, dies aber eher die Plattform der Zielgruppe sein dürfte (zumindest mehr als Facebook) sollte auch die Betrachtung der GIF-Fähigkeit eines Spiels in die Entwicklung einfließen. Je mehr Farbverläufe, je schneller die Bildfolge, je kleinteiliger das Spiel ist, umso weniger wirkt es in einer GIF-Animation. Das Problem kann sogar bei Videoaufnahmen entstehen. Bei nahezu allen YouTube-Videos wirkt CYPEST bei weitem nicht so, wie es tatsächlich ist, weil die Schatten zu verwaschen, die Partikeleffekte kaum richtig zu erkennen sind. Was im tatsächlichen Spiel richtig cool wirkt, geht im Video weitestgehend unter. Stattdessen sieht man in den Videos nur, wie der Spieler alle paar Sekunden stirbt.

In der dritten Folge dieser kleinen Serie schreibe ich darüber, wie man die Tonaufnahmen bei eigenen Spielen optimiert.

Date Created
5. Juli 2018
Author
sven