

Exotisches Interlacing



by scharfis_brain

Inhalt

1. Einführung in den Aufbau von Video

1.1 Erklärung der halbbildweisen Übertragung von PAL-Video

1.2 Erklärung der halbbildweisen Videoaufnahme in einer Kamera (normales Interlacing)

1.3 Erklärung der benutzten Notationen in diesem Dokument

2. Exotisches Interlacing

2.1 Telecine mit PAL-Speedup

2.1.1 Kein Interlacing

2.1.2 Phase shifted

2.1.3 Dynamischer Phase Shift

2.2 Telecine ohne PAL-Speedup

2.2.1 Ohne Blending

2.2.1a Progressive

2.2.1b Interlaced

2.2.2 Mit Blending

2.3 Normgewandeltes Filmmaterial

2.3.1 Erklärung der NTSC-Telecine per 3:2-Pulldown

2.3.2 Ohne Blending

2.3.3 Mit Blending

2.3.4 Wie eine korrekte NTSC-->PAL-Wandlung aussehen sollte

2.4 Was die TV-Sender noch so alles auf Lager haben

2.4.1 Speedup von 25fps auf 26fps, um mehr Werbung zeigen zu können

2.4.2 Slowdown von 25fps auf 22.5 fps um Zensur-Cuts zeitmässig zu verbergen

3. Schlußbemerkungen, Danksagungen

4. Anhang

4.1 AviSynth-Script zur Simulation der Effekte

Dieser Text widmet sich dem Interlacing (und dessen Behandlung/Beseitigung), welches eigentlich nicht auftreten dürfte, wenn Filme ordentlich auf das PAL-Format gemastert würden. Es wird der grundsätzliche Umgang mit AVISynth und Grundwissen bzgl. Interlacing vorausgesetzt, ich werde darauf garnicht bzw. nur kurz eingehen.

Jeder, der schonmal mit Video am PC gearbeitet hat, wird sicherlich bemerkt haben, dass es bei manchen

Videsequenzen zu horizontalen Streifen bei bewegten Inhalten im Video kommt. Dies ist der berüchtigte Interlacing-Effekt. Ich werde diese horizontalen Streifen im weiteren Text nur noch als Kämmе oder Interlacing bezeichnen.

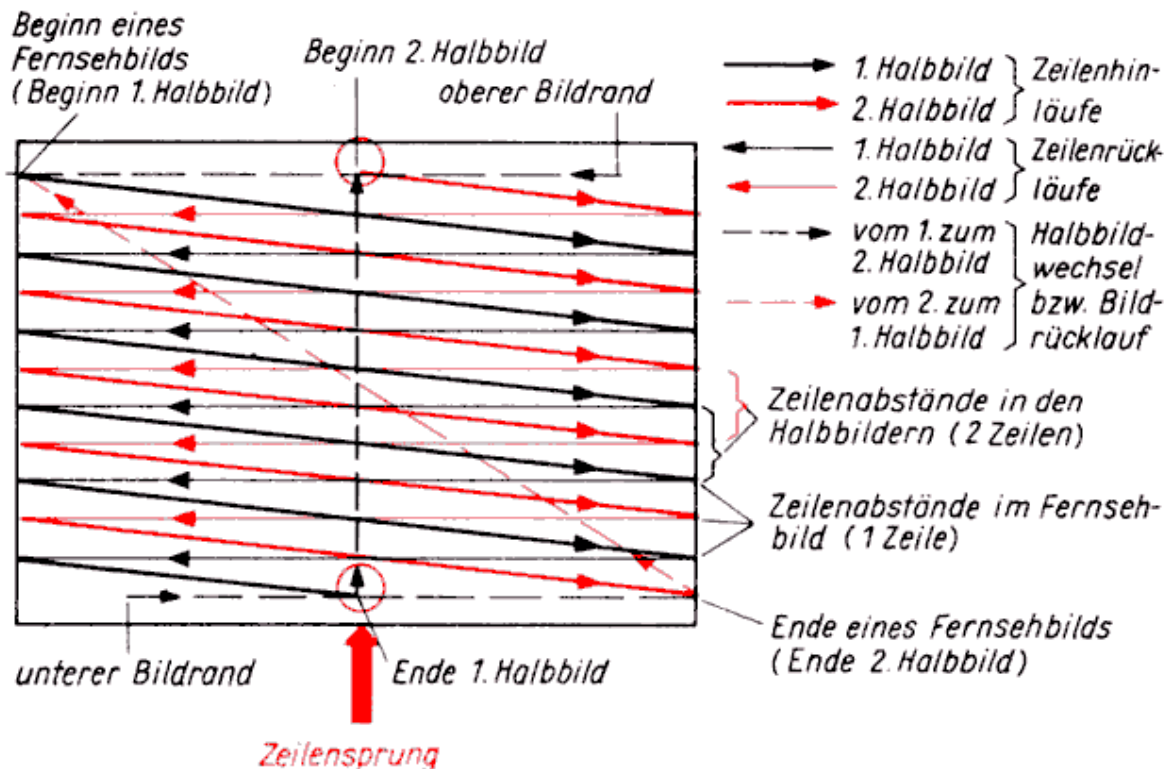
Nun, wie entstehen diese Kämmе?

1. Einführung in den Aufbau von Video

1.1 Erklärung der halbbildweisen Übertragung von PAL-Video

Dafür müssen wir wissen, wie ein analoges Video übertragen wird. Bei dem in unserem Gebiet üblichen Fernsehstandard PAL haben wir eine vertikale Auflösung von 576 bildrelevanten Zeilen. Das komplette Bild wird dabei 25 mal in der Sekunde übertragen. Da es aber unerträglich auf einem Fernseher flimmern würde, wenn man das Bild nur 25 mal in der Sekunde auf dem Bildschirm komplett neuzeichnen würde, haben sich kluge Leute in den zwanziger Jahren das Zeilensprungverfahren (Interlacing) ausgedacht. Bei diesem Verfahren werden zuerst die **geradzahligen Zeilen** des Bild übertragen (also die **Zeilen 2,4,...,572,576, erstes / unteres Halbbild** bzw. **Even / Bottom Field**) und erst danach die **ungeradzahligen Zeilen (1,3,5,...,573,575, zweites / oberes Halbbild** bzw. **Odd / Bottom Field**)¹. Dadurch verringert man das Flimmern, da man nun eine Bildwiederholfrequenz von 50Hz hat. Dies erkaufte man sich aber mit einer Auflösungsreduktion in bewegten Bereichen des Videos. Beim Fernsehen fällt dies bei sich bewegenden diagonalen Kanten auf, sie weisen immer einen Treppcheneffekt auf. Statische Bildelemente hingegen haben volle Detailauflösung. Das Zeilensprungverfahren ist also eine Art Kompression.

¹)Da ein analoges Halbbild exakterweise nur 575 (anstatt der digitalisierten 576) Bildzeilen hat, beinhaltet jedes Halbbild auch ‚nur‘ 287 und eine **HALBE** Zeile. Das komplette PAL-Bild hat demzufolge 574 komplette Zeilen und je eine halbe Zeile **oben (Zeile 1)** und **unten (Zeile 576)**. Siehe dieses Bild:



1.2 Erklärung der halbbildweisen Videoaufnahme in einer Kamera (normales Interlacing)

In einer Videokamera werden nicht 25 Vollbilder pro Sekunde aufgenommen, sondern 50 Halbbilder pro Sekunde, d.h. jedes Halbbild enthält einen anderen Bewegungszustand als das vorhergehende. Deswegen sind zwei aufeinanderfolgende Halbbilder auch nicht mehr deckungsgleich. Wenn wir nun solch ein Video am PC bearbeiten, wird uns dieser immer zwei Halbbilder gleichzeitig anzeigen, da aber beide Halbbilder, aufgrund der verschiedenen Zeitindexe, nicht deckungsgleich sind, gibt es die bekannten Kämmе. Dazu ein Halbbildschema:

```
Videofield1  0  2  4  6  8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50
Paarung n.mö.  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´  ´
Videofield2  1  3  5  7  9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49 51
Zeit          -----interlaced----->
```

Dies wäre das normale Interlacing, welches sich entweder durch deinterlacing beseitigen lässt, oder besser noch ohne deinterlacing als interlaced MPEG zu speichern ist, da es hierbei zu keinem Qualitätsverlust kommt.

1.3 Erklärung der benutzten Notationen in diesem Dokument

Dies ist eine typische Halbbildarstellung, wie ich sie des öfteren benutzen werde:

```
Filmframe    A  B  C  D  E          A  B  C  D  E  F  G  H
Videofield1  A  B  B  D  E          A  B BC D  DE F  FG GH
Paarung      |  | /  ´  /          `  |  ´  |  ´  |  ´  /
Videofield2  A  B  C  E  F          AB B C D  E  F  G  H
Zeit         --p--p--i--i--i-->    -i--p--i--p--i--p--i--i-->
```

Filmframe kennzeichnet die individuellen Einzelbilder des originalen Films, VOR aller Konvertierung.

Videofield1&2 beschreibt die Halbbilder des vorliegenden Videostroms. Entweder

- in sequentieller Aufnummerierung 0,2,...,2n für field1 und 1,3,...,2n+1 für field2
oder
- mit Filmframe-Notation: Einzelne Buchstaben kennzeichnen, dass das Halbbild nur den Inhalt des Filmframes trägt. Während zwei aufeinanderfolgende Buchstaben einen Blend, also eine Mischung zweier Filmframes kennzeichnet.

Paarung beschreibt die Gleichheit einzelner Halbbilder zueinander:
| oder / weist auf 2 Halbbilder pro Vollbild hin (Normalfall)
| / oder | \ weist auf drei Halbbilder pro Filmframe hin (passiert bei NTSC-Telecine).
, oder ´ weist auf ein verwaistes Halbbild pro Filmframe hin, ist bei "echtem" Interlacing jedoch normal, da es hierbei 50 hfps gibt.

Zeit beschreibt den Zeitverlauf.
p kennzeichnet ein Progressive Frame (nur möglich, wenn zwei gleiche Halbbilder übereinander stehen)
i kennzeichnet ein Interlaced-Frame

2. Exotisches Interlacing

Das exotische Interlacing entsteht nun, wenn jemand einen Film auf das PAL-Format konvertieren möchte. Dieses Verfahren wird Telecine genannt. Ein Film hat immer eine Bildwiederholfrequenz von 24 fps. PAL hingegen hat 50 hfps (half frames per second). Jetzt haben wir mehrere Möglichkeiten, den Film auf das PAL-Format zu konvertieren.

2.1 Telecine mit PAL-Speedup

2.1.1 kein Interlacing

Die meistgenutzte und beste Methode ist der sogenannte PAL-Speedup. Hierbei wird der Film von 24fps auf 25fps beschleunigt. Dann wird er auf PAL konvertiert, indem immer zwei Halbbilder den gleichen Filmframe beinhalten. Diese Beschleunigung hat aber einen kleinen Nebeneffekt: Die Tonhöhe ändert sich um nahezu einen Halbton nach oben. Bei aufwändigeren Wandlungen wird dies durch ein komplexes Verfahren vermieden. AVISynth bringt ab der Version 2.5.5 eine Funktion TimeStretch mit, die den Speedup/Slowdown ohne Tonhöhenänderung vornehmen kann.

```
Filmframe      A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
-----
Videofield1   A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Paarung       | | | | | | | | | | | | | | | | | | | | | | | |
Videofield2   A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Zeit          -----progressive----->
```

```
Filmframe A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Topfield  A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Bottomfield A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
PC-View  A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
```

Diese Sequenz würde am PC kein Interlacing bemerkbar machen, da der PC immer die Halbbilder 0&1, 2&3, 4&5 gleichzeitig darstellt und somit auch nur individuell die Filmframes A, B und C.

Dieses Video kann problemlos progressive behandelt werden.

Nachahmung in AviSynth:

```
AVISource("24fpsFilm.avi")
AssumeFPS(25,true)
```

Behebung in AVISynth:

-nicht notwendig!

2.1.2 Telecine mit PAL-Speedup - Phase shifted

Manchmal aber passiert es, dass sich diese Halbbildsequenz um 1/50 Sekunde verschiebt. Das dazugehörige Schema sähe so aus:

```
Filmframe      A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
-----
Videofield1   * A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Paarung       / / / / / / / / / / / / / / / / / / / / / / / /
Videofield2   A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Zeit          -----interlaced----->
```

```
Filmframe A B C D E F G H I J K L M N O P Q R S T U V W X Y
Topfield  A B C D E F G H I J K L M N O P Q R S T U V W X Y
Bottomfield A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
PC-View  A A B B C C D D E E F F G G H H I I J J K K L L M M N N O O P P Q Q R R S S T T U U V V W W X X Y Y
```

Wir wir jetzt sehen können, beinhalten die Halbbilder 0&1, 2&3, 4&5 jetzt auch zwei verschiedene Filmframes (nämlich A&B;, B&C;, C&D;), da der PC, wie oben beschrieben, die Halbbilder immer nur in diesem Schritten

darstellt, hat jedes Bild des Films Kämme, ohne dass der Film tatsächlich interlaced wäre, weil immer zwei Film-Frames gleichzeitig dargestellt werden. Am TV fällt das aber in keinster Weise auf.

Nachahmung in AviSynth:

```
AVISource("24fpsFilm.avi")
AssumeFPS(25,true).DoubleWeave().SelectOdd()
```

Diese Art der Telecine wird als Phase shifted oder manchmal auch als Perverse Telecine bezeichnet und lässt sich mit Hilfe von AVISynth recht einfach beheben, indem man die Halbbilder neu sortiert:

Behebung in AviSynth:

```
DoubleWeave().SelectOdd()
```

2.1.3 Telecine mit PAL-Speedup - dynamischer Phase Shift

Teilweise passiert es auch, dass sich normale Telecine und perverse Telecine abwechseln (wie z.B. die Independence Day - DVD). Dies ließe sich nicht mehr durch DoubleWeave().SelectOdd() beheben, da dann die normale Telecine zur perverse Telecine wird. Hierbei kann also nur noch eine automatische Lösung helfen, und zwar in Form von Telecide aus Donald Grafts Decomb Package:

```
Telecide(order=1,guide=2,post=0)
```

Telecide() sucht hierbei im Video nach ähnlichen Halbbildern und fügt diese dann zusammen (sorgt also für eine korrekte Sortierung der Halbbilder), so dass das Video wieder progressive wird.

2.2 Telecine ohne PAL-Speedup

2.2.1 ohne Blending

2.2.1a progressive

Hierbei wird der Film nicht auf 25fps beschleunigt, sondern nur jede Sekunde ein Filmbild 2x (Vollbild) respektive 4x (Halbbild) gezeigt:

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y			
Videofield1	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	X	Y		
Paarung																												
Videofield2	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	X	Y		
Zeit	-----progressive----->																											

rot = Duplikate

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	
Topfield	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	X	Y
Bottomfield	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	X	Y
PC-View	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	X	Y

Das 24. Filmbild wird immer 4x gezeigt, und zwar in den Halbbildern 46 bis 49. Dieses Verfahren führt allerdings zu sekundlichen Rucklern.

Nachahmung in AVISynth:

```
AVISource("24fpsFilm.avi")
ChangeFPS(25)
```

oder

```
AVISource("24fpsFilm.avi")
SelectEvery(24,0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,23)
```

Behebung in AviSynth:

```
Decimate(25)
```

Die doppelten Bilder können mit Decimate(x) aus dem Decomb Package entfernt werden. Decimate sucht nach doppelten Vollbildern im Video und entfernt dann alle x Bilder das verdoppelte Bild. Das resultierende Video hat dann eine Framrate von 24fps, also genau die des Films.

2.2.1b interlaced

Um das Ruckeln, welches bei 2.2.1a entsteht, zu verringern, fügt man nicht sekundlich ein Vollbild ein, sondern 2 mal pro Sekunde ein Halbbild. Das führt aber dazu, dass man alle 12 Filmframes einen phase shift (und einen leichten Ruckler) bekommt:

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y		

Videofield1	A	B	C	D	E	F	G	H	I	J	K	L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	
Paarung													/	/	/	/	/	/	/	/	/	/	/	/	/		
Videofield2	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	X	Y	
Zeit	-----progressive-----												-----interlaced--												----- ---p--->		

rot = Duplikate

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	
Topfield	A	B	C	D	E	F	G	H	I	J	K	L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
Bottomfield	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	X	Y
PC-View	A	B	C	D	E	F	G	H	I	J	K	L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Das Video ist also im Wechsel 13 Frames lang progressive und dann 12 Frames interlaced.

Nachahmung in AVISynth:

```
AVISource("24fpsFilm.avi")
ChangeFPS(50).SeparateFields().SelectEvery(4,0,3).Weave()
```

Behebung in AVISynth:

Zuerst muss der phase shift beseitigt werden. Dazu benutzen wir wieder

```
Telecide(order=1,guide=2,post=0)
```

Durch Telecide() erhält man ein Video, wie in 2.3.1a., welches dann mittels

```
Decimate(25)
```

wieder auf 24fps gebracht wird

2.2.2 Telecine ohne PAL-Speedup - mit blending - interlaced

Dies ist eine der grausamen Methoden, Film auf PAL zu transferieren. Hierbei werden Halbbilder, die zeitlich nicht genau unter einem Filmframe liegen, aus den zeitlich benachbarten Filmbildern gemischt (blending). Die Sequenz könnte so aussehen (je nach Konverter etwas anders):

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z		
Videofield1	A	B	C	D	E	F	G	H	I	J	K	L	LM	MN	NO	OP	PQ	QR	RS	ST	TU	UV	VW	WX	XY	Y	Z	
Paarung n.mö.																												
Videofield2																												
Zeit	-----interlaced----->																											

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Topfield	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Bottomfield	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
PC-View	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Die **fett** markierten Halbbilder sind intakte Halbbilder, deren Inhalt dem eines einzelnen FilmFrames entspricht. Die anderen Halbbilder sind blends, d.h. ihr Inhalt besteht aus der Mischung zweier benachbarter Filmframes.

Nachahmung in AviSynth:

```
AVISource("24fpsFilm.avi")
ConvertFPS(50).SeparateFields().SelectEvery(4,0,3).Weave()
```

Behebung in AVISynth:

Am TV fällt eine solche Telecine fast garnicht auf. Am PC hingegen hat man in fast jedem Frame interlacing. Auch greifen hier keine Standardmethoden, wie Telecide zur Beseitigung des Interlacings, da die benachbarten Halbbilder nichtmehr ähnlich sind. Man koennte das Video deinterlacen, um dem Interlacing beizukommen, aber das würde mehr als die Hälfte der Bilder zu blends machen:

```
FieldDeinterlace(order=1, blend=false)
```

Ich habe eine Lösung gefunden, die möglicherweise Abhilfe schaffen kann:

Zuerst muss man das Video mit einem Fullframerate-Deinterlacer, wie z.B. DGBob, Kernelbob oder TMCKernelBob auf 50 fps deinterlacen. Danach muss man mit Hilfe von Unblendpattern die intakten Halbbilder auswählen lassen. Dies setzt allerdings voraus, dass die Blend-Sequenz ein konstantes Muster aufweist. Es dürfen auch keine Framedrops im Video sein, denn dadurch wird das Blend-Pattern auch verschoben.

```
KernelBob(4)
UnblendPattern(offset,24,false,false)
```

2.3 Normgewandeltes Filmmaterial

2.3.1 Erklärung der NTSC-Telecine per 3:2 Pulldown

Um einen Film auf NTSC (60 hfps) zu bringen, kann man ihn keinem Speedup von 24 auf 30 fps unterziehen, das würde zu sehr auffallen. Anstatt dessen zeigt man die Filmframes im Wechsel 2x und 3x, wie die Sequenz verdeutlicht:

Filmframe	A	B	C	D	E	F	G	H	I	J	K			
Videofield1	A	B	B	C	D	E	F	F	G	H	I	J	J	K
Paarung		/	/	/		/	/	/		/	/			
Videofield2	A	B	C	D	D	E	F	G	H	H	I	J	K	L
Zeit	--p--p--i--i--p--p--p--i--i--p--p--p--i--i-->													

Oder eine etwas andere Darstellungsweise:

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z		
Videofields	0	2	5	7	10	12	15	17	20	22	25	27	30	32	35	37	40	42	45	47	50	52	55	57	60	62		
	1	3	6	8	11	13	16	18	21	23	26	28	31	33	36	38	41	43	46	48	51	53	56	58	61	63		
		4		9		14		19		24		29		34		39		44		49		54		59		64		
Zeit	----->																											

Dadurch kommt es zu einem Dauerruckeln, welches zwar bemerkbar, aber nicht allzu störend empfunden wird.

Nachahmung in AviSynth:

```
AVISource("24fpsFilm.avi")
SelectEvery(2,0,0,0,1,1).SeparateFields().SelectEvery(4,1,2).Weave()
```

Den 3:2 Pulldown kann man mit Hilfe von AVISynth und dem Decomb-Package wie folgt rückgängig machen:

```
Telecide(order=1,post=0)
Decimate(5)
```

Dieses Verfahren heisst *inverse Telecine* (IVTC)

2.3.2 Normgewandeltes Filmmaterial - ohne Blending

Dies ist die einfachste, billigste und schlechteste Methode, eine Normwandlung durchzuführen. Hierbei wird das Video zuerst mit einem Fullframerate-Deinterlacer auf 60 fps gebracht und dann jedes 6. Bild entfernt. Das führt zu einem ganz unerträglichem 10 Hz ruckeln und bei Filmmaterial zu folgender Sequenz:

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z		
Videoframes	0	2	x	7	10	12	15	x	20	22	25	27	30	32	x	37	40	42	45	x	50	52	55	57	60	62		
	1	3	6	8	x	13	16	18	21	x	26	28	31	33	36	38	x	43	46	48	51	x	56	58	61	63		
		4		9		14		19		24		x	34		39		44		49		54		x		64			
Zeit	----->																											

x - gelöschte Bilder

Danach werden die Vollbilder wieder auf Halbbildgröße reduziert.

Die Resultierende PAL - Sequenz:

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z		
Videofields	0	2	5	6	9	10	13	15	17	19	21	23	25	27	30	31	34	35	38	40	42	44	46	48	50	52		
	1	3		7		11	14	16	18	20	22	24	26	28		32		36	39	41	43	45	47	49	51	53		
		4		8		12						29		33		37										54		
Zeit	----->																											

oder in Halbbildabfolge:

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
Videofield1	0	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40	42	44	46	49	51	53
Paarung		/	,	/	,	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	
Videofield2	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39	41	43	45	48	50	52	54
Zeit		--p--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--	p--i--

(` und , kennzeichnen verwaiste, aber bildrelevante Halbbilder)

Filmframe **A B C D E F G H I J K L M N O P Q R S T U V W X Y**

Topfield **A B B D D F F G H I J K L M N O P Q R S T U V W X Y**

Bottomfield **A B C D E F G H I J K L M N N P P R R S T U V W X Y**

PC-View **A B C D D F F G G H H I I J J K K L L M M N N O O P P Q Q R R S T U V W X Y**

Wie man sieht, ergibt das ein hübsches Durcheinander in der Bildabfolge. Sieben (G-M) und sechs (S-Y) Frames sind immer komplett durch zwei Halbbilder abgeschlossen. Aber dann kommt eine Folge von fünf Frames, von denen drei (BDF) Frames aus je drei Halbbildern und zwei (CE) Frames aus nur einem einzigen Halbbild bestehen.

Nachahmung in AviSynth:

```
AVISource("30fpsTelecinedFilm.avi")
DGBob(order=1).Changefps(50).Separatefields().Selectevery(4,0,3).Weave()
#oder
DGBob(order=1).Selectevery(6,0,1,2,3,4)
Separatefields().Selectevery(4,0,3).Weave()
```

Behebung in AviSynth:

Mit Telecide kann man bei einem solchen Stream leider gar nichts anfangen, da Telecide die verwaisten Halbbilder einfach "schluckt", d.h. aus dem vorhergehenden Dreifachhalbbild werden 2 Vollbilder (4 Halbbilder).

Zurzeit kann dem aber man recht gut mit

```
SmartDecimate(24,50,tel=0.9)
```

Herr werden...

Allerdings gibt es hin- und wieder mal Ruckler, wenn SmartDecimate() die verwaisten Halbbilder (B,C,O,Q) nicht korrekt erkennt.

2.3.3 Normgewandeltes Filmmaterial - mit Blending

Hierbei wird aus dem, auf Fullframrate deinterlacedtem, Video nicht einfach jedes 6. Bild gelöscht, sondern die Bilder werden über ein gewichtetes Blending (Mischen) von 60fps auf 50fps reduziert:

Blending allgemein:

NTSC-Field	1	2	3	4	5	6	7	8	9	10	11	12	13						
Wichtung in %	100	75	25	60	40	40	60	25	75	100	75	25	60	40	40	60	25	75	100
PAL - Field	1	2	3	4	5	6	7	8	9	10	11								
Zeit	----->																		

Blending bei Telecined Film:

NTSC-Field	A	A	B	B	B	C	C	D	D	D	E	E	F
			/ \	/ \	/ \				/ \	/ \	/ \		

Wichtung in %	100	75	25	60	40	40	60	25	75	100	75	25	60	40	40	60	25	75	100		
		\	/		/	\		/	\		/	\		/	\		/	\			
PAL - Field	A	AB	B	B	CB	C	D	D	DE	E	F	F	FE	E	DE	D	CD	C	B	AB	A
Zeit	----->																				

Für eine ganze Sekunde:

(übereinanderliegende Buchstaben kennzeichnen einen Blend)

NTSC : A A B B C C D D E E F F G G H H I I J J K K L L M M N N O O P P Q Q R R S S T T U U V V W W X X Y Y Z Z

PAL : A A B B C C D D E E F F G G H H I I J J K K L L M M N N O O P P Q Q R R S S T T U U V V W W X X Y Y Z Z

B C E G H I J L N O Q S T U V X Z

In Halbbildnotation:

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
Videofield1	A	B	BC	D	DE	F	FG	GH	HI	IJ	J	KL	L	MN	N	O	P	Q	RS	S	T	U	V	W	X	Y	Z
Paarung	`		,		,		,		,		,		,		,		,		,		,		,		,		,
Videofield2	AB	B	C	D	E	F	G	H	I	J	K	L	M	N	NO	P	PQ	R	RS	ST	TU	UV	V	WX	X	YZ	Z



Dies ist nur ein theoretisches Pattern, da bei einer echten Konvertierung zwischen 59.94fps und 50fps gewandelt werden muss. Dadurch verschiebt sich das Pattern alle 1000 NTSC-Frames und es gibt mehr blended-fields als dieses Beispiel zeigt. Nichtsdestotrotz sollte dieses Beispiel zeigen, wie schlimm sich eine solche Normwandlung auf Film auswirkt!

Nachahmung in AviSynth:

```
AVISource("24fpsFilm.avi")
SelectEvery(2,0,0,0,1,1).SeparateFields().SelectEvery(4,1,2).Weave()
DGBob(order=1).ConvertFPS(50).SeparateFields.SelectEvery(4,0,3).Weave()
```

Behebung in AviSynth:

Wenn man versucht, ein solches Video zu deinterlacen, bekommt man 2 Probleme, je nach Deinterlacer:

1. es Ruckelt 1x Pro Sekunde unerträglich. Das passiert wenn man mit einem Deinterlacer im 'interpolate'-Modus deinterlaced. FieldDeinterlace(blend=false)
2. kein Ruckeln, aber dafür sind fast alles Frames blends. Dies passiert, wenn der Deinterlacer die Halbbilder mischt: FieldDeinterlace(blend=true)

Es gibt zwar keine wirkliche Lösung, aber mehrere Lösungsansätze:

1. Restore24 von Dideé verwenden:

```
AVISource("normwandel.avi")
restore24()
```

2. Kernelbob, Unblend und Smartdecimate, die Einfachausführung von restore24:

```
AVISource("normwandel.avi")
KernelBob(4)
```

```
Unblend()
SmartDecimate(24,100)
```

3. Unblendpattern verwenden. Macht aber nur Sinn, wenn das Video nicht allzu viele Offsetsprünge hat. StarTrek o.ä. ließe sich damit wohl kaum wiederherstellen, der Aufwand wäre einfach viel zu groß:

```
AVISource("normwandel.avi")
KernelBob(4)
UnblendPattern(offset,24,false,false)
```

2.3.4 Wie eine korrekte NTSC -> PAL Wandlung für Film aussehen sollte

Eine "gute" Normwandlung sollte die Film-Parts eines Videos einer inverse Telecine (IVTC, Kap. 2.3.1.) unterziehen. Die IVTC stellt die ursprünglichen 24fps des Files wieder her. Dann wird das gesamte Video auf 62.5 fps beschleunigt und dann der IVTC-te Part OHNE Blending auf 25 fps beschleunigt und nur die Video-Parts werden per Blending auf 50hfps konvertiert. Dies vermeidet die im vorherigen Abschnitt behandeltem Probleme.

Die AVISynth-Funktion converter.avs beherrscht diese Art der Normwandlung. Wahlweise mit oder ohne Speedup. Film-Parts werden jedoch in jeden Falle ohne Blending konvertiert.

2.4 Was die TV-Sender noch so alles auf Lager haben

Hier landen alle anderen Misshandlungen, denen Film-Streams noch so zum Opfer fallen können.

2.4.1 Speedup von 25fps auf 26fps, um mehr Werbung zeigen zu können

Vor einiger Zeit hatte BaronVlad ein interlacing-Problem mit einem Film, der auf Sat.1 lief. Nach einigen (Fern-) Analysen, die für mich nur in Verwirrung ausarteten, erhielt ich ein Sample von ihm. Nach eingehender Analyse war ich erschrocken. Ich fand 26, in Worten: sechszwanzig, Filmbilder pro Sekunde in 50 Halbbildern pro Sekunde, die restlichen 24 Halbbilder waren blended:

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	

Videofield1	A	B	C	D	E	F	G	H	I	J	K	L	MN	NO	OP	PQ	QR	RS	ST	TU	UV	VW	WX	Y	Z		
Paarung n.mö.																											
Videofield2	AB	BC	CD	DE	EF	FG	GH	HI	IJ	JK	KL	M	N	O	P	Q	R	S	T	U	V	W	X	YZ	ZA		
Zeit	-----interlaced----->																										

Filmframe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
Topfield	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
Bottomfield	A	B	D	E	F	G	H	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A		
PC-View	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A

Dieser Stream veranlasste mich, Unblendpattern() zu schreiben, womit der Baron diesen Clip erfolgreich wieder Progressive bekam.

Sat.1 beschleunigte diesen (über 2h Netto langen) Film von schon 25fps (PAL-Speedup) auf 26fps, was alles in allem wohl noch einen weiteren Werbeblock von einigen Minuten Dauer zuließ.

2.4.2 Slowdown von 25fps auf 22,5 fps um Zensurcuts laufzeitmäßig zu verbergen

Dies hab ich bei einem Sample von Ashyak vom Sender Pro7 gesehen. Hierbei wurde der Slowdown glücklicherweise ohne Blending durchgeführt, sodass man alle 10 Bilder ein Duplikat hat. Dieses Pattern ist dem aus 2.2.1b ähnlich, nur dass hierbei die Tripel öfter auftreten:

```

Filmframe      A B C D E F G H I J K L M N O P Q R S T U V W X
-----
Videofield1   A B C D E E F G H I J K L M N O O P Q R S T U V W X
Paarung       | | | | | / / / / / | | | | | / / / / | | | | |
Videofield2   A B C D E F G H I J J K L M N O P Q R S T T U V W X
Zeit          -p--p--p--p--p--i--i--i--i--i--p--p--p--p--p--i--i--i--i--p--p--p--p--p-->

```

rot = Tripel

```

Filmframe  A B C D E F G H I J K L M N O P Q R S T U V W
Topfield   A B C D E E F G H I J K L M N N O P Q R S T U V W W
Bottomfield A B C D E F G H I I J K L M N O P Q R R S T U V W
PC-View    A B C D E F G H I J K L M N O P Q R S T U V W

```

Prinzipiell hätte Telecide(post=0).Decimate(10) geholfen. Da der Film aber vermutlich nicht durchgängig mit 22.5 fps lief, hätte man ihn Szenenweise analysieren müssen.

3. Schlussbemerkungen, Danksagungen

Na? Kopfschmerzen bekommen? Aspirin hilft!

Ich sollte hiermit eigentlich fast alle Film-Misshandlungen abgehandelt haben. Solltet ihr dennoch etwas finden, was sich nicht mit meinen Erklärungen deckt, postet euer Problem bei <http://forum.gleitz.info> !

Mehr zu meinen Funktionen gibt's hier:

kernelbob & unblendpattern <http://forum.doom9.org/showthread.php?s=&threadid;=66093&>
 converter.avs <http://forum.doom9.org/showthread.php?s=&threadid;=67161&>

Vielen Dank an alle, die mir Ihre Sampleclips zur Verfügung gestellt haben:

BaronVlad 26fps Speedup
Ashyak 22,5fps Slowdown und Normwandlung ohne blending
i4004/ivo telecine ohne PAL-Speedup mit blending
Manono div. NTSC-blended-Clips
Der Karl StarTrek **ist** eine Normwandlung! Und wehe Dir, Du streitest noch mal mit mir darüber! ;-)
Dogleg
Encod3r
kortduorsch Anime-Normwandlung
narler dynamischer Phase shift
wim
zeronegative

zisoft für die HTML-Umsetzung

uvm, die mir gerade nicht einfallen.

4. Anhang

4.1 AviSynth-Script zur Simulation der Effekte

Das folgende AviSynth-Script erzeugt eine Video-Sequenz, mit der die besprochenen Effekte simuliert werden können:

```
src=BlankClip(1,256,384,"RGB24",24)
rot=$FF0000
gruen=$00FF00
blau=$0000FF
gelb=$FFFF00
lila=$FF00FF
tuerkis=$00FFFF
gr=320
off=48
A=src.SubTitle("A",size=gr,text_color=rot)
B=src.SubTitle("B",x=off,size=gr,text_color=gruen)
C=src.SubTitle("C",size=gr,text_color=blau)
D=src.SubTitle("D",x=off,size=gr,text_color=gelb)
E=src.SubTitle("E",size=gr,text_color=lila)
F=src.SubTitle("F",x=off,size=gr,text_color=tuerkis)
G=src.SubTitle("G",size=gr,text_color=rot)
H=src.SubTitle("H",x=off,size=gr,text_color=gruen)
I=src.SubTitle("I",size=gr,text_color=blau)
J=src.SubTitle("J",x=off,size=gr,text_color=gelb)
K=src.SubTitle("K",size=gr,text_color=lila)
L=src.SubTitle("L",x=off,size=gr,text_color=tuerkis)
M=src.SubTitle("M",size=gr,text_color=rot)
N=src.SubTitle("N",x=off,size=gr,text_color=gruen)
O=src.SubTitle("O",size=gr,text_color=blau)
P=src.SubTitle("P",x=off,size=gr,text_color=gelb)
Q=src.SubTitle("Q",size=gr,text_color=lila)
R=src.SubTitle("R",x=off,size=gr,text_color=tuerkis)
S=src.SubTitle("S",size=gr,text_color=rot)
T=src.SubTitle("T",x=off,size=gr,text_color=gruen)
U=src.SubTitle("U",size=gr,text_color=blau)
V=src.SubTitle("V",x=off,size=gr,text_color=gelb)
W=src.SubTitle("W",size=gr,text_color=lila)
X=src.SubTitle("X",x=off,size=gr,text_color=tuerkis)
Y=src.SubTitle("Y",size=gr,text_color=rot)
Z=src.SubTitle("Z",x=off,size=gr,text_color=gruen)

res=a+b+c+d+e+f+g+h+i+j+k+l+m+n+o+p+q+r+s+t+u+v+w+x+y+z

res=res.crop(0,88,0,-40)

res+res+res+res+res
```