PRESTO – Preservation Technologies for European Broadcast Archives

IST-1999-20013

Deliverable 5.3: High-Quality Format Conversion Software

DOCUMENT IDENTIFIER   PRESTO-T43-JRS-20020507 Format Conversion

DATE   May, 7th, 2002

ABSTRACT

AUTHOR, COMPANY   W. Bailer, G. Kienast, H. Mayer, P. Schallauer (JRS)

INTERNAL REVIEWER

WORKPACKAGE / TASK   Task 5.3

DOCUMENT HISTORY

Release	Date	Reason of change	Status	Distribution
0.1	2002-04-10	Document created & initial input	Draft	Confidential
1.0	2002-05-07	Document finalized	Final	Confidential

Table of Contents

1. Introduction

This section gives a brief technical overview on the idea of a digital film master with its advantages and drawbacks for archival and broadcast usage.

1.1. Workflow

The basic idea is to hold beside or instead of analogue film a single digital film master, which can be converted on demand to the required video or film output formats. The main advantage for broadcast usage is that cost intensive multiple scans are avoided. The main advantage for archival usage is the long term digital preservation of film.

To be able to convert a single digital master to all intended output formats (i.e. SDTV, HDTV, film reprint) the minimum scan quality is given by the highest quality output format. Maximum useful scan quality can be derived from stock properties (e.g. for 35mm original negative about 8 Mega pixels with about 12 bits per channel logarithmic quantisation). To avoid quality decrease due to the scanning process the sampling structure of the digital film master is equivalent to the original film sampling structure (i.e. 24p, 18p).

All output formats are derived from the single digital film master. Several output format properties have to be taken into account, proper handling of interlaced and progressive formats with different field/frame rates, spatial and densitometric resolution and aspect ratio (i.e. letter box). There are solutions for format conversion available on the market, but they are either limited in spatial or densitometric resolution, in conversion quality or in system throughput. A software solution has no limitation in spatial and densitometric resolution, can provide high conversion quality because of no limitation of algorithmic complexity (i.e. motion estimation) and has non-realtime system throughput.

1.2. Colour Treatment

Two different approaches for handling colours are possible. Colour-corrected master and non colour-corrected master. When taking the approach of a colour-corrected master during the scanning process either global or shot by shot colour correction is applied. This is a standard procedure for broadcast usage. The goal is to get a digital master which is as close as possible to the analogue film at shooting time.

The approach of a non-colour corrected master is based on the requirement to get a digital equivalent of the current analogue film, which contains all colour degradations of the current film. A non colour-corrected master requires higher densitometric resolution, otherwise digital colour correction in a post processing step after output format conversion would not be possible in high quality.

The decision which approach is best is a film-ethical, a workflow and cost issue. The colour-corrected master approach avoids additional costs for colour correction after each output format conversion.

Beside the two colour correction approaches for the digital master there is also the requirement of taking into account different colour spaces and gamma properties of the different output formats. Colour space and gamma property is defined standardized for each output format (i.e. for NTSC, PAL, HDTV, computer, film) and thus the necessary conversion can be applied automatically during the format conversion step.

1.3. Storage Format

Currently the DPX[1] file format has become an industry standard for film production. It supports resolution independency (spatial and densitometric), but does not meet the requirements of broadcasters because, for instance, it does not contain sound data or metadata. Film is stored as a number of single uncompressed image files.

The MXF format intends to overcome these limitations by, for instance, support of metadata and by support of film structure information (shot information). Standardisation is currently in progress (see http://www.pro-mpeg.org/mxf.htm).

---

[1] SMPTE 268M-1994: File Format for Digital Moving-Picture Exchange (DPX)

2. Functional Description of the Format Conversion Software

2.1. Overview

The high-quality format conversion software is capable of converting film or video material by changing one or more of the following properties:

- spatial resolution

- aspect ratio

- temporal sampling rate (frames/fields per second), with both motion-compensated and non-motion compensated conversion methods

- temporal sampling structure (interlaced, progressive) and field order of interlaced material, including elimination of 3:2 and 2:2 pulldown

- color space and gamma value

Spatial resolution

Any spatial resolution of input and output images is supported, there are no limitations for conversions.

Aspect ratio

Any aspect ratio is supported and there are no limitations for aspect ratio conversion. A region of interest (ROI) can be specified for aspect ratio conversion. The output frame is generating by scaling the ROI of the input frame to the specified output size.
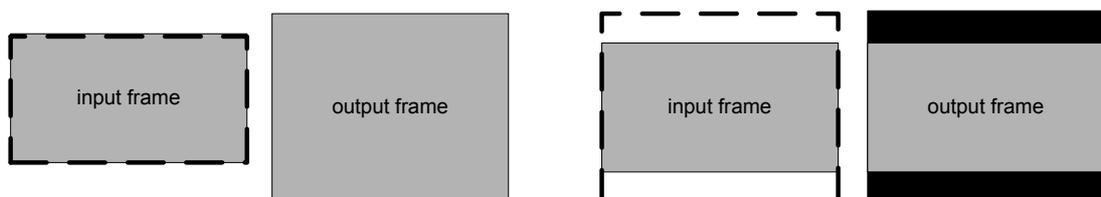
The figure shows possible ROIs (dashed) for 16:9 to 4:3 aspect ratio conversion. On the left, the entire input image is scaled non-uniformly to the output resolution. On the right, the ROI is larger than the input image and the undefined parts will be filled black (letter box).

Temporal sampling rate

The format conversion software converts between arbitrary temporal sampling rates. Changing the sampling rate requires in most cases the interpolation of frames or fields at temporal positions, where there are no frames/fields in the input sequence. In the presence of camera or object motion, motion compensated interpolation is required for high-quality conversion. Both motion-compensated and non-motion compensated conversions are supported, they are described in detail in 2.2.

Temporal sampling structure

The format conversion software can handle both progressive and interlaced material and convert between both types. For best possible quality, interlaced input is de-interlaced using motion compensation. The resulting progressive sequence is the basis for conversion to any progressive or interlaced output format.

Apart from "real" interlaced material, conversion methods like 3:2 and 2:2 pulldown produce interlaced material, which contains two or three fields that were generated from the same frame. As the fields are not equidistantly distributed along the time axes, the sequence may not be treated as interlaced material. Instead, the format conversion software reconstructs the original progressive sequence from the input and uses it as if it were a progressive input.

Gamma correct color space conversion

Different video standards use different color spaces, i.e. the red, green and blue primaries have different values. Conversion between the standards includes therefore also conversion between the color spaces. The color space conversion can only be done with respect to the gamma values of input and output sequence. Of course it is also possible to use the conversion software for gamma correction without changing the color space. Details on gamma correct color space conversion are described in section 2.3.

The format conversion software supports both 8 and 16 bit color depth per channel with any number of color channels (usually one or three).

2.2. Supported Conversions

The following table lists the different conversions supported by the format conversion software.

Input	Output	m.c.[2]	Description	Example
n p	m p	y	motion compensated progressive to progressive conversion with arbitrary sampling rates, if n is an integer multiple of m motion compensation has not advantages and is therefore turned off	film upconversion from 18 fps to 24 fps
n p	m p	n	non-motion compensated progressive to progressive conversion, faster than motion-compensated conversion, but causes superimposition defects on moving objects if temporal n is not an integer multiple of m	48 fps to 24 fps
n p	n p	n	special case of progressive to progressive conversion without change of the temporal sampling structure, i.e. only change of size and/or aspect ratio	24 fps 2k to 24 fps SDTV resolution
n p	m i	y	motion-compensated progressive to interlaced conversion with arbitrary sampling rates	24 fps film to PAL
n p	m i	n	non-motion compensated progressive to interlaced conversion, faster than motion-compensated conversion, but causes superimposition defects on moving objects if temporal n is not an integer multiple of m	60 frames ps to 60 fields ps

---

[2] motion compensated

n p	2*n* i	n	In contrast to the non-motion compensated conversion described before, which produces temporally equidistant output, *n* p to 2*n* i conversion is often done by creating 2 frames from each field (2:2 pulldown). With the input and output data formats used in this application, this conversion does not make sense, as it the difference between input and output is just the interpretation of an input frame as one frame or two nested fields. Input and output of this conversion would be identical and it is therefore not supported.	24 frames ps to 48 fields ps
n p	5/2*n* i	n	The 3:2 pulldown is a classic non-motion compensated method to generate 60 fields (NTSC) from 24p (film). It produces alternately two or three fields from each frame.	24 frames ps to 60 fields ps
n i	*m* p	y	motion compensated de-interlacing and conversion, if the input is the result of a pulldown, the original progressive sequence is first reconstructed	50 fields ps to 24 frames ps
n i	*n* p	y	motion compensated de-interlacing only, if the input is the result of a pulldown, the original progressive sequence is first reconstructed	50 fields ps to 25 frames ps
n i	*m* p	n	non-motion compensated de-interlacing (see below) and conversion (cf. *n* p to *m* p conversion), if the input is the result of a pulldown, the original progressive sequence is first reconstructed	60 fields ps (resulting from 3:2 pulldown) to 24 frames ps
n i	*n* p		non-motion compensated deinterlacing, by default, median de-interlacing is used, which might cause defects on moving objects, alternatively spatial de-interlacing can be used, which provides only a reduced vertical resolution	
n i	*m* i	y	motion compensated de-interlacing and conversion of the resulting progressive sequence to an interlaced sequence with different sampling structure, if the input is the result of a pulldown, the original progressive sequence is first reconstructed	50 fields ps to 60 fields ps
n i	*m* i	n	non-motion compensated de-interlacing and conversion of the resulting progressive sequence to an interlaced sequence with different sampling structure, if the input is the result of a pulldown, the original progressive sequence is first reconstructed	
n i	*m* i	n	non-motion compensated interlaced to interlaced conversion without prior de-interlacing but temporal interpolation between same type of fields (even-even and odd-odd), can only be applied if the image size does not change	

Table 2.1: Supported Conversions.

Examples are only mentioned in cases, where the conversion method can be applied without loss of quality.

2.3. Gamma Correct Color Space Conversion

The format conversion software uses configuration files defining color spaces and gamma profiles.

2.3.1. Color Space Definitions

Color spaces are defined with respect to the CIE XYZ color space. Each specific RGB color space is defined by it's white point (the XYZ values of the color white) and the relative amounts of x=X/X+Y+Z and y=Y/X+Y+Z of the primaries R, G and B.

The following table lists the color spaces for which definitions are available (ID is the name used to specify a color space as command line parameter):

Name(s)	ID	x_R	y_R	x_G	y_G	x_B	y_B	white
EBU Tech.3213-E and SMPTE RP145-1994	EBUTech3213E	0.64	0.33	0.29	0.60	0.15	0.06	D65
NTSC VIQ	NTSC_VIQ	0.67	0.33	0.21	0.71	0.14	0.08	C
SMPTE-240M	SMPTE240M	0.67	0.33	0.21	0.71	0.15	0.06	D65
SMPTE-C	SMPTE_C	0.630	0.340	0.310	0.595	0.155	0.070	D65
ITU-R BT.709-4	ITUR_BT709	0.640	0.330	0.300	0.600	0.150	0.060	D65

Table 2.2: Color space definitions.

The white points are defined as follows:

Name	Definition	X	Y	Z
C (average sunlight),	2°, CIE 1931	98.074	100	118.232
D65 (Daylight)	2°, CIE 1931	95.047	100	108.883

Table 2.3: White point definitions.

Other color spaces can be used by adding appropriate color space definition files. The file format is described in 3.2.2.

2.3.2. Gamma Profile Definitions

The input material is assumed to have a gamma >1, so that a correction with a gamma <1 will be performed to linearize it. After color space conversion, a correction with a gamma >1 will be applied.

There are two types of gamma functions supported by the format conversion software. An exponential only function given by

$$g(x) = x^{gamma}$$

and a piecewise linear function

$$g(x) = \begin{cases} k \cdot x, x < a \\ p \cdot x^{gamma} - q, x \geq a \end{cases}$$

which is a generalized version of the gamma function in ITU-R BT.709 and SMPTE-170M.

The following gamma profiles are available in the format conversion software (ID is the name used to specify a color space as command line parameter):

Name	ID	gamma	a	k	p	q
default	default	1.0	-	-	-	-
Gamma 0.45	gamma045	0.45	-	-	-	-
SMPTE-170M	SMPTE170M	0.45	0.018	4.5	1.099	0.099

Table 2.4: Gamma profile definitions.

Other gamma profiles can be used by adding appropriate gamma profile definition files. The file format is described in 3.2.2.

3. Technical Description of the Format Conversion Software

3.1. Overview

The format conversion software consists of two modules: A preprocessing module preparing the image sequence for motion estimation and resampling and the actual sampling structure conversion module, which generates the image of the output sequence.

The format converter provides a motion compensated and a non-motion compensated mode. Except for the last two steps, the sampling structure conversion module is different depending on the mode. Figure 3.1 shows the structure of the application for the motion compensated mode, Figure 3.2 the sampling structure conversion module for non-motion compensated mode.
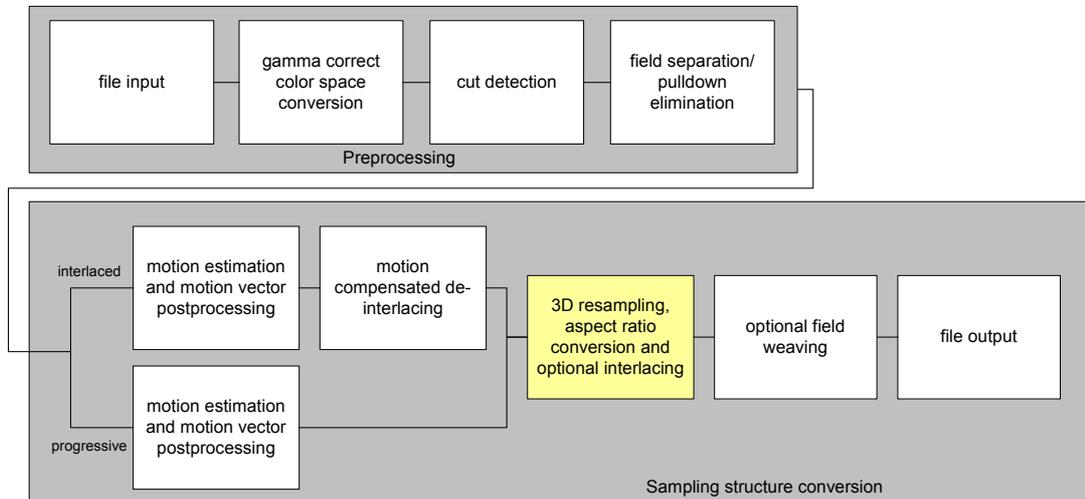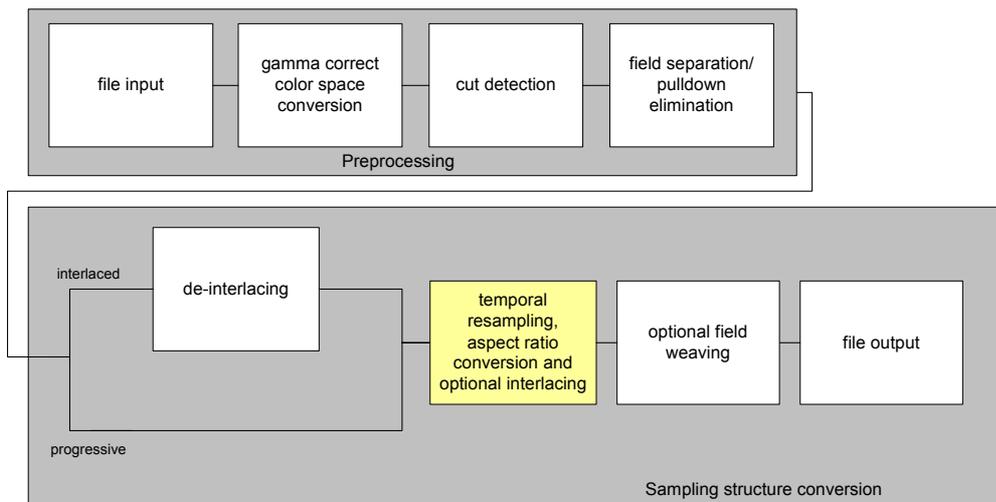

Figure 3.1: Block diagram of motion compensated mode.

Figure 3.2: Block diagram of non-motion compensated mode.

3.2. Preprocessing

The preprocessing module performs tasks that are necessary for the motion estimation and/or the resampling step.

3.2.1. File Input

The image data is read as a sequence of progressive frames. Interlaced data is represented as woven fields, two fields are stored in a frame. The image sequence metadata is filled from user input or during automatic detection steps, at this point, the spatial resolution of the source sequence is read from the image files and stored in the image sequence metadata.

The input frame will be cropped to the specified input ROI, the other modules will just get the ROI and the frame size is adjusted to the ROI's size. In motion compensated mode, the ROI will be enlarged by a percentage specified in the parameters to make motion estimation easier.

If the spatial output resolution is smaller than the input ROI and the input material is progressive, the frames will be scaled down for efficiency reasons before any further processing is done. In this case spatial lowpass filtering will be applied before downsampling.

The following file formats are supported:

TIFF, PNM, SGI, RGB, DPX

3.2.2. Gamma Correct Color Space Conversion

Gamma correct color space is performed after a frame is read to make sure that all frames have been processed when another step needs to access this frames. Interlaced frames are still woven in this step, as this reduces the amount of data to be processed.

The gamma correct color space conversion consists of three steps:

- inverse input gamma correction
- color space conversion
- output gamma correction

Both color space definitions and gamma profiles are read from files. The formats are as follows (cf. section 2.3).

Color Space definitions

Color space definitions are stored in text files named *ID.colorspace*, where ID is the name used to reference it as command line parameter. The following sample defines the NTSC VIQ color space.

```
# color space definition NTSC_VIQ      comment
#
xr=0.67                                x of red in XYZ
yr=0.33                                y of red in XYZ
xg=0.21                                x of green in XYZ
yg=0.71                                y of green in XYZ
xb=0.14                                x of blue in XYZ
yb=0.08                                y of blue in XYZ
# white point CIE C, 2°
Xwhite=98.074                          X value of white point
```

```
Ywhite=100.000                          Y value of white point
Zwhite=118.232                          Z value of white point
```

Gamma Profiles

Gamma profile definitions are stored in text files named *ID.gamma*, where ID is the name used to reference it as command line parameter. The following sample defines the SMPTE-170M gamma profile.

```
# piecewise linear gamma function (SMPTE-170M)
#                                       comment
gamma=0.45                              exponent of gamma function
isLinear=1                              0...exponential only, 1...piecewise linear
k=4.5                                   coefficient of linear part
a=0.018                                 value under which linear function is used
p=1.099                                 coefficient of exponential part
q=0.099                                 to be subtracted from exponential part
```

3.2.3. Cut Detection

The cut detection finds hard cuts in the image sequence in order to treat each shots separately and avoid interpolating over shot boundaries.

In the case of interlaced input, the cut detection has to be able to recognize cuts between the two woven fields stored in a frame.

3.2.4. Field Separation/Pulldown Elimination

Corrects the temporal sampling structure in the case of interlaced or 2:2/3:2 pulldown input. The information about the material type comes from user input. The results of the cut detection will be used in the case of 2:2/3:2 pulldown material, as cuts may disturb the regular order.

The different types of material are treated internally as follows.

Interlaced

The two fields are extracted and written into separate frames depending on the field order with the remaining lines left undefined until de-interlacing.

2:2 Pulldown

In a 2:2 pulldown sequence, an original frame is represented by two fields, which can either be stored in one frame or across two frames. As the two fields belonging together originate from the same frame, the matching fields are put together.

The sequence is treated as progressive from now on.

3:2 Pulldown

In a 3:2 pulldown sequence, an original frame has been used to produce alternately two or three fields. Two or three matching fields will be detected and put together as one frame. The order may change at a cut.

The sequence is treated as progressive from now on.

Both the 2:2 and the 3:2 pulldown elimination will fail on damaged sequences, if the correspondence between two fields can not be recognized because of defects.

3.3. Sampling Structure Conversion

3.3.1. Motion Compensated Mode

3.3.1.1 Motion Estimation and Motion Vector Postprocessing

The result of the motion estimation and motion vector postprocessing step is a dense motion vector field (i.e. a motion vector for each pixel) from one frame to the following including reliability information and an occlusion map (areas being covered or uncovered between the two frames).

An hierarchical motion estimation method, the Hierarchical Feature Vector Matcher (HFVM)[3] is used.

Motion estimation is always performed one frame in advance, as this information is needed for motion-compensated de-interlacing and may also be required for motion extrapolation.

Motion estimation on interlaced input is done between two adjacent fields and the vertical offset and reduced resolution of the estimate is corrected.

There may be areas for which motion cannot be estimated reliably (e.g. because of lack of structure). Motion vectors from these areas are marked as unreliable to avoid their use in the resampling step.

Spatio-temporal interpolation may not be done in covered or uncovered areas, as there is no motion trajectory for pixels in these areas. These regions are marked during the motion estimation process.

3.3.1.2 Motion Compensated De-interlacing

This is applied to interlaced input only.

De-interlacing is done using image information from the field to be de-interlaced and the previous and next field. In the previous and next field the lines missing in the field to be de-interlaced are defined, and motion information between them, too The result is a frame, which will be treated as progressive input in the resampling step.

If no temporal interpolation will be carried out after de-interlacing (i.e. in a *n* interlaced to *n* progressive conversion), the de-interlacing will be performed in the resampling step.

3.3.1.3 3D Resampling, Aspect Ratio Conversion and Optional Interlacing

The 3D resampling steps generates the frames/fields of the output sequence. Based on the estimated motion vector fields, image data from adjacent frames is projected to the image plane of the image to be interpolated, taking into account spatial scale factors and the ROI. The samples projected onto the interpolation plane will be unequally distributed and sometimes contradictory, so an approximation model that also takes into account the different reliability of motion vectors, is used to generate an image approximating the available samples. The samples come from three sources:

- from sufficiently reliable motion vectors

---

[3] G. Paar and W. Pölzleitner: Descent and Landing Phase: Vision Based Spacecraft Motion Estimation and Elevation Modelling. Proc. Int. Symp. on Missions, Technologies and Design of Planetary Mobile Vehicles, Toulouse, Sept. 1992

- from extrapolated motion vectors

- from fallback (that is a only temporal interpolation for areas without sufficient reliable motion information)

The approximation is iteratively refined and an image is generated. Optionally (for interlaced output sequences), only the required lines of the image are calculated.

3.3.1.4   Optional Field Weaving

This is applied to interlaced input only.

If the output image sequence is interlaced, two successive fields are woven into one frame before writing the output to disk.

3.3.1.5   File Output

The output image sequence is written to disk. The same formats as for file input are supported.

3.3.2.   Non-Motion Compensated Mode

3.3.2.1   De-Interlacing

This is applied to interlaced input only.

In non-motion compensated mode, de-interlacing can be done either only spatially, which has advantages if much motion is present, or using a vertical-temporal median filter providing a better vertical resolution but is endangered of defects in the presence of motion.

A conversion without spatial scaling between two interlaced formats can also be done without de-interlacing; then temporal interpolation is done between fields of same type in the resampling step.

3.3.2.2   Temporal Resampling, Aspect Ratio Conversion and Optional Interlacing

The images of the input image sequence are scaled to the required resolution, if the images have to be downscaled, lowpass filtering is applied before subsampling (in the case of progressive input downscaling has already been done in the input step to avoid handling a large amount of data). The output image is generated by temporal interpolation from the input images taking the ROI into account. Optionally (for interlaced output sequences), only the required lines of the image are calculated.

3.3.2.3   Optional Field Weaving

This is applied to interlaced input only.

If the output image sequence is interlaced, two successive fields are woven into one frame before writing the output to disk.

3.3.2.4   File Output

The output image sequence is written to disk. The same formats as for file input are supported.

4.   Usage

4.1. Usage of the Conversion Software

4.1.1. Input and Output Data

Both input and output image sequences are expected to be stored as a number of files in the same directory in one of the following image formats: TIFF, PNM, SGI, RGB, DPX. Interlaced image sequences are expected to be stored as a sequence of images containing two woven fields. The filename is assumed to be in the format *name.n.ext*, where n is the frame number with up to 8 digits and ext is the file extension. The number of digits in the frame number has to be constant for all frames of one sequence.

The output image sequence produced by the converter will be in the same directory as the input and the filename will be *name.con.n.ext*.

4.1.2. Command Line Syntax

The conversion software is implemented as a command line tool. It can be called from the command line using the following syntax:

converter [options] *<path- and filename> <start frame> <end frame>*

where

- options are described below,
- *path- and filename* contains an absolute or relative path and the name of the files of the image sequence, but not the frame number and the extension,
- *start frame* is the number of the first frame in the sequence to be processed and
- *end frame* is the number of the last frame in the sequence to be processed.

Options			
Name	**Description**	**Accepted Values**	**Default**
Description of input sequence[4]			
-infps *fps*	frames/fields per sec. of input seq.	decimal number >0	24
-intempstruct *s*	sampling structure of input sequence	one of the values: PROG progressive IL interlaced PD32 3:2 pulldown PD22 2:2 pulldown	PROG
-in_even	(interlaced input only) set this option, if input is lower field first (equivalent to even, e.g. NTSC)	n/a	not set (upper first = odd = PAL)

---

[4] Width, height and number of channels of the input sequence are determined automatically from the first frame.

-roi_x *x*	process region of interest (ROI), specifies x coordinate of upper left corner of the ROI in the input sequence	-2^31<x<2^31, x=1 is the left border of the input image sequence	1
-roi_y *x*	process region of interest (ROI), specifies y coordinate of upper left corner of the ROI in the input sequence	-2^31<y<2^31, y=1 is the left border of the input image sequence	1
-roi_w *w*	process region of interest (ROI), specifies width of the ROI in the input sequence	1<w<2^31, w=0 sets w to the width of the input image	0
-roi_h *h*	process region of interest (ROI), specifies height of the ROI in the input sequence	1<h<2^31, h=0 sets h to the height of the input image	0
-align *nr*	alignment frame in input sequence, a frame of the output sequence is aligned temporally exactly to this frame	start frame <= nr <= end frame	
Description of output sequence			
-outW *width*	width of output images in pixel	integer values 1<width<2^31	720 (SDTV)
-outH *height*	height of output images in pixel	integer values 1<height<2^31	576 (SDTV)
-outfps *fps*	frames/fields per sec. of output		
-outtempstruct *s*	sampling structure of output sequence	one of the values: PROG progressive IL interlaced PD32 3:2 pulldown PD22 2:2 pulldown	PROG
-out_even	(interlaced input only) set this option, if input is lower field first (equivalent to even, e.g. NTSC)	n/a	not set (upper first = odd = PAL)
Algorithmic options			
-non_mc	do not use motion compensation	n/a	
-nodeinterl	(interlaced input only) do not de-interlace in a non-motion compensated interlaced to interlaced conversion without scaling, but interpolate between same types of fields (i.e. between even fields and between odd fields)	n/a	
-iter *n*	number of iterations during resampling	1<n<5	2
-spatialresf *f*	function used for spatial resampling (cubic and gauss provide better quality than linear, cubic can only be used if spatial scale factor is around 1, gauss can be used with any scale factor)	one of LINEAR CUBIC GAUSS	GAUSS

-tempresf *f*	function used for temporal resampling (linear is recommended, cubic and gauss for further investigation of subjective quality)	one of LINEAR CUBIC GAUSS	LINEAR
-relthresh <n>	reliability threshold (do not use motion vectors with less reliability)	[0.0;1.0]	0.5
-extrapolate	extrapolate motion vectors from adjacent frames if necessary (not for interlaced input), NOTE: not recommended	n/a	
-fallback	use fallback mode (i.e. spatial interpolation if there is no reliable motion information), NOTE: recommended	n/a	
-occlfallback	use occlusion information in fallback mode, NOTE: not recommended	n/a	
-spatialNMCDI	(interlaced input only) use only spatial de-interlacing in non-motion compensated mode	n/a	not set, median de-interlacing is used
Color space conversion			
-colorCorrection	apply gamma correct color space conversion, requires specification of input and output color space and gamma profiles	n/a	
-inColorspace *name*	name of input color space	name of color space definition file (without extension)	EBUTech3213E
-outColorspace *name*	name of output color space	name of color space definition file (without extension)	EBUTech3213E
-inGamma *name*	name of input gamma profile	name of gamma profile definition file (without extension)	default (gamma=1.0)
-outGamma *name*	name of output gamma profile	name of gamma profile definition file (without extension)	default (gamma=1.0)

Table 4.1: Command line options.

5. Results

Figure 5.1: Non-motion compensated (left) and motion compensated (right) conversion. The superimposition defects on the moving head are clearly visible in the non-motion compensated image, while the moving regions are still sharp in the motion compensated image.

Figure 5.2: The frames from the original sequence which were used for interpolation.

6.   Glossary

fps   frames per second

i   interlaced

p   progressive

ROI   region of interest