Configuration

Description

Configuration information passed to the warping library is used to check constraints and as basis for computations.
Following is a description of the warping configuration parameters:

Reference Points

The reference points are represented by a 2D array with coordinates inside the display area to be warped. There are **NumOfRefPointsX** in horizontal direction and **NumOfRefPointsY** in vertical direction, both having values in the range 0..255. These are equally distributed over Bitmap-pixel.
Example:

- NumOfRefPointsX: 21 (0x15)
- NumOfRefPointsY: 11 (0x0B)

The number of reference points used influence to output of the distorted image with regard to Bitmap size. The output image will be less jagged the higher the number of reference points are used for warping computation

PixResolution

The Resolution of warping parameters determinine the warping accuracy. The final displacement of the reference points is the product of warping parameter by parameter resolution. Therefore, a warping parameter is given by the desired displacement divided by resolution.

The warping library supports the following pixel:

- 0: $^{1}/_{16}$ px
- 1: $^{1}/_{8}$ px
- 2: ¼ px
- 3: ½ px
- 4: 1 px
- 5: 2 px
- 6: 4 px
- 7: 8 px

Example:
If a texture point at (0,0) should be mapped at (10,10) the corresponding X[0] and Y[0] coordinates

can be calculated by dividing the desired displacement by pixel resolution.
For a pixel resolution of $^{1}/_{16}$ px (0.0625):

```
X[0] = Xnew[0] – Xold[0] / 0.0625 = 160 (0xA0)
Y[0] = Ynew[0] –Yold[0] / 0.0625 = 160 (0xA0)
```

For a pixel resolution of ¼ px (0.25):

```
X[0] = Xnew[0] – Xold[0] / 0.25 = 40 (0x28)
Y[0] = Ynew[0] –Yold[0] / 0.25 = 40 (0x28)
```

NumOfBitsPerParam

Represents the number of bits used for each parameter (n-bit representation = bit length of each parameter).
The value should be in the range is 1..16.
Example:

- If NumOfBitsPerParam is 8 the warping parameters corresponding to X[0]Y[0] are represented as 0xA0 0xA0.
- If NumOfBitsPerParam is 12 the warping parameters corresponding to X[0]Y[0] are represented as 0x0A 0x00 0xA0.

> The **NumOfBitsPerParam** and **PixResolution** are represented on 8 bits, upper three bits are used for NumOfBitsPerParam and lower bit for PixResolution.

Example::

- 0x96 -> NumOfBitsPerParam: 12, PixResolution 0 ($^{1}/_{16}$ px)
- 0x98 -> NumOfBitsPerParam: 12, PixResolution 2 (¼ px)

DisplaySizeX and DisplaySizeY

Width and height data (in pixel) of the display hardware (range: 0..65535).

Example:

- 0x01 0xE0 – DisplaySizeX – 480px width
- 0X00 0XF0- DisplaySizeY – 240 px height

BitmapAreaX and BitmapAreaY

Width and height data (in pixels) of source surface used as input for warp. The reference points are equally distributed over this range in x and y direction with values in the range 0..65535.

Example:

- 0x01 0x98 – BitmapAreaX – 408px width
- 0X00 0XBE- BitmapAreaY– 190 px height

> Condition:
> BitmapAreaX≤ DisplaySizeX
> BitmapAreaY≤DisplaySizeY

BitmapAreaCenter

The Bitmap Area Center Point describes the center point of the image area used by ABK.

BitmapAreaShift

The Bitmap Area Shift describes the connection vector between the origin of the bitmap coordinate system (0,0) and the center point of the bitmap area - i.e. the Bitmap Area Center Point.

BitmapAreaShiftX and BitmapAreaShiftY

Shift values of bitmap area center point in horizontal and vertical direction (range 0..65535).

- ”0” BitmapAreaShiftX means the application window starts at the left edge on the display.
- ”0” BitmapAreaShiftY means the application window starts at the upper edge on the display.

Example:

- 0x00 0xCC – BitmapShiftAreaX – 204px
- 0X00 0X5F- BitmapShiftAreaY– 95 px

> Condition:
> (BitmapAreaY/2+BitmapAreaShiftY≤DisplaySizeY) AND (BitmapAreaY/2–

> BitmapAreaShiftY ≥0
> (BitmapAreaX/2+BitmapAreaShiftX≤DisplaySizeX) AND (BitmapAreaX/2–BitmapAreaShiftX ≥0

IlluminatedDisplayAreaX and IlluminatedDisplayAreaY

Width and height data (in pixel) for illuminated area. This represents the region in which the warped surface will be rendered. If the source surface size extends beyond the IlluminatedDisplayArea it will be scaled down with regards to display size.

Example:

- 0x01 0xE0 – IlluminatedDisplayAreaX – 480px width
- 0X00 0XF0- IlluminatedDisplayAreaY– 240 px height

> Condition:
> IlluminatedDisplayAreaX ≤ DisplaySizeY
> IlluminatedDisplayAreaY ≤ DisplaySizeY

IlluminatedDisplayAreaShiftX and IlluminatedDisplayAreaShiftY

Shift values of illuminatedDisplay area center point in horizontal and vertical direction (range 0..65535).

Example:

- 0x00 0xF0 – BitmapShiftAreaX – 240px
- 0X00 0X78- BitmapShiftAreaY– 120 px

> Condition:
> (IIIDisplayAreaX/2+IIIDisplayAreaShiftX≤DisplaySizeX) AND (IIIDisplayAreaX/2–IIIDisplayAreaShiftX ≥0)
> (IIIDisplayAreaY/2+IIIDisplayAreaShiftY≤DisplaySizeY) AND (IIIDisplayAreaY/2–

> IllDisplayAreaShiftY ≥0)

RotationValue

The rotation value is represented in grad *100 with values in the range -18000 .. +18000. This is the default rotation value which will be applied to the warp image.

The rotation takes place around the warped center point of the bitmap area used. With an uneven number of reference points in x- and y-direction, this point coincides with the middle reference grid point, and the coordinates of the rotation axis are present. If an even number of reference grid points is used, the coordinates of the warped center point of the image are be calculated.

Example:

- 0x01 0xF4 – RotationValue: 500/100 = 50

ParamSet

Set of reference point shift vectors column-ordered.

The warping parameter correspond to the shift vectors (in pixels) of warping interpolation points in warped images in comparison to reference points in non-warped images (divided by the specified resolution). To save memory space and computing time, the absolute coordinates are not saved as warping parameters, but rather only the distance from the reference points (without rotation and shifting).

ParamSetLength

Length of ParamSet in bytes. This is calculated by:

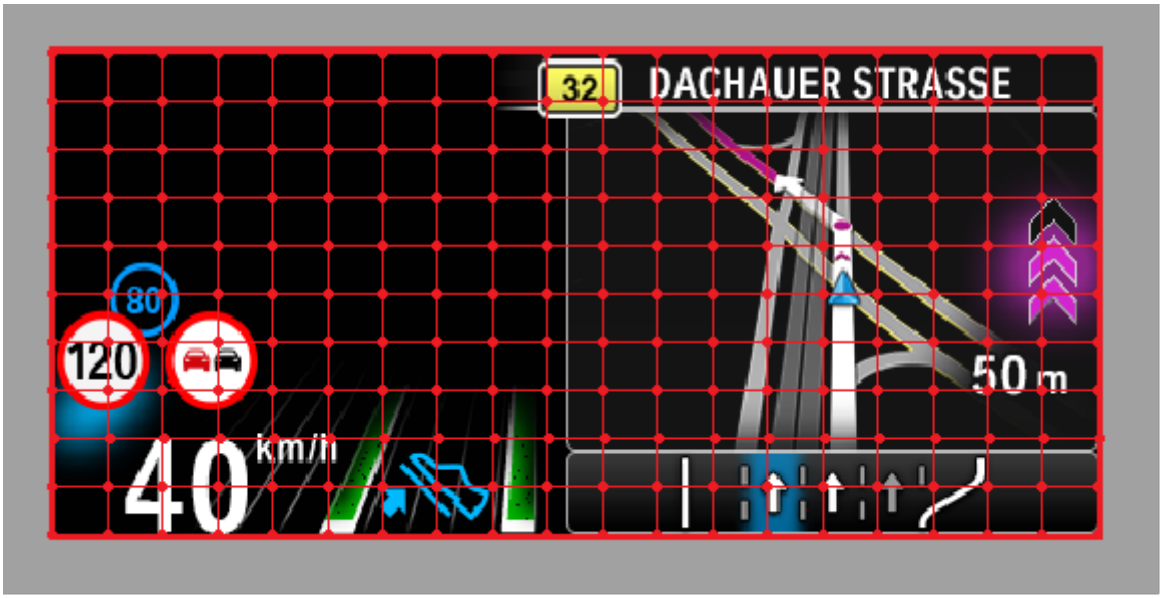
```
ParamSetLength = NumRefPointsX * NumRefPointsY * 2 * NumBitsPerParam
```

RotationLimit

This value represents the maximal rotation angle in grad – both positive and negative values are accepted. If the RotationLimit value is chosen such that at maxim rotation point, the image will extend beyond display size, it will be scaled down to the size that fits inside the display area when a rotation equal to RotationLimit is applied.

The *display area* comprises the whole image including the grey border.
The *bitmap image* area is the area within the grey border. This is the area we are interested in and which should be warped.
The reference points are evenly distributed over the bitmap image area.

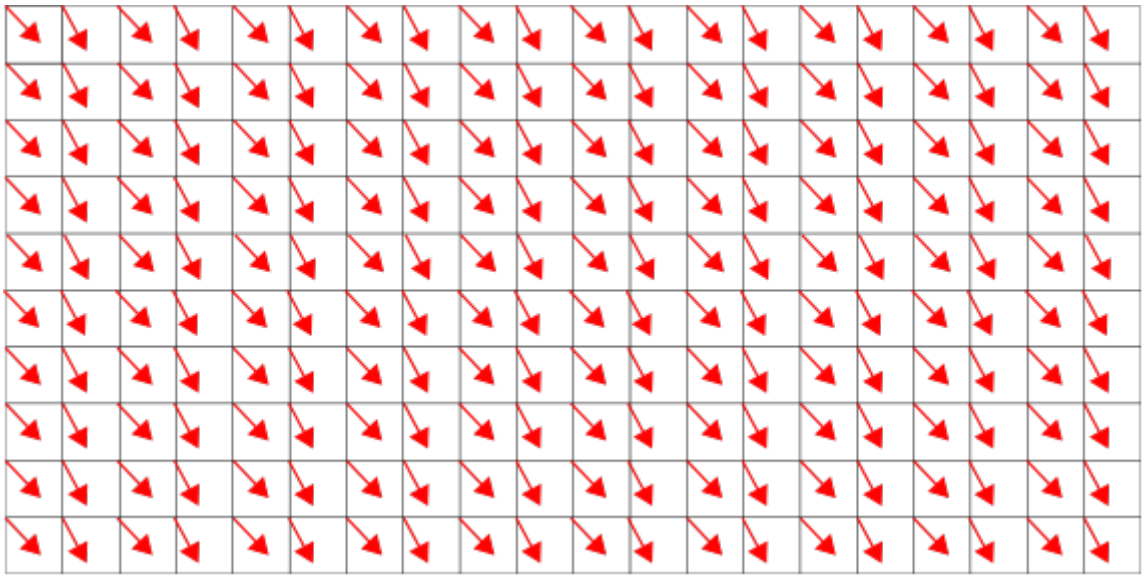
> The illuminated display area is the part of the display, which is backlit and projected to the windshield.



Reference Point Shift Vectors

The warping parameters are based on reference point. These points are mapped by a 2D array containing coordinates inside the display area to be warped. For ++-each reference point, a shift vector is stored by the application. Every shift vector determines the amount of distortion to be applied to the corresponding point in x- and y-direction. Figure 2 shows a reference point grid with corresponding shift vectors.

To preserve memory, the shift vectors are stored in a compressed format. The format is determined by configuration information passed to warping library (number of bits used, value resolution) and is used by warping library to de- and encode shift vectors. The decoded shift vectors are cached in the warping library.



Warp Matrix

The warp matrix is computed by the warping library. It is a 2-dimensional matrix. Each point corresponds to a reference point. The value of each point is the vector-addition of the reference point and corresponding shift vector. All values of the warp matrix are normalized to the range [0..1].

Warp Image Bounds

Warp image bounds can be computed by the warping library. Warp image bounds comprise upper left corner point of the bitmap image area, its width, and height (normalized to the range [0..1]).

They are used to specify to Candera which part of the rendered image shall be mapped to the warping mesh.

Revision #6
Created 2023-02-16 07:00:55 UTC by Tsuyoshi.Kato
Updated 2026-08-24 07:40:37 UTC by Kanai Tomoaki