Overscan correction with xrandr

Overscan

Old and cheap TVs have no way to adjust the overscan. When connected to the computer, it will be as if the TV is slightly zoomed in on the display, leaving around 5% of the display off the screen.

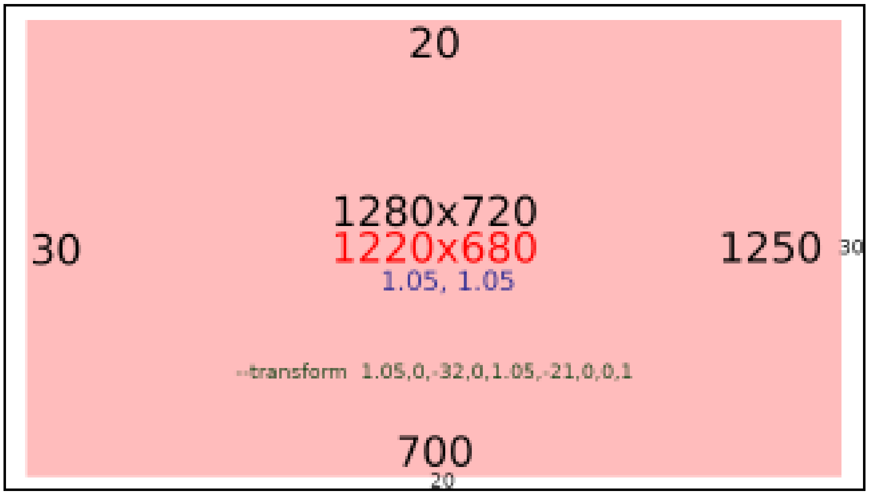
Using a python tool

If the TV with overscan is positioned at the top left (eg `xrandr --output TV --left-of LVDS-1`), the provided `xrandr_transform.py` will prompt for the output name and mouse clicks to get the bounds. The output will be a `xrandr` command to correct the overscan.

```
xrandr --output HDMI-1 --mode 1280x720 \
 --transform 1.05,0,-35,0,1.07,-27,0,0,1 --pos 0x0
```

Illustration

As far as I know, the actual size of the overscan varies by TV. I measured using gimp.



reproduce

1. open gimp with new image set to resolution (e.g. 1280x720)
2. fullscreen (F11), toggle tools (TAB), menubar, status, etc in view
3. draw selection over visible part of the screen and fill
4. zoom out and measure

```
xcf2png 1280p.xcf |
 convert - -scale 25% -bordercolor black -border 1x1 1280p_border.png
```

Correction Calculation

Ultimately, we want what is red in the above figure to cover the full image. That is, we want to map the bounds of what is displayed to the reported resolution. `xrandr` can handle this with a transform matrix. (See pdf for rendered latex.)

The matrix

The transform calculation is

$$\begin{bmatrix} x_{out} \\ y_{out} \\ z \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_{in} \\ y_{in} \\ z \end{bmatrix} \tag{1}$$

This display on this TV screen

- With an x border of 30 on both sides, we have a final width of $1280 - 2 * 30 = 1220$.
- Our original width is 1280, or $\frac{1280}{1220} = 1.05$ times greater than the TV's effective width.
- In the new space we need to move $30 * 1.05 = 31.5$ pixels.
- With the y border of 20: $\frac{720}{720-2(20)} = 1.05$ times greater than the TV's effective height.
- In the new space, y should be shifted $20 * 1.05 = 21$
- We are not doing any rotation, x and y are independent of one another, so $b = e = 0$.

This matrix

$$\begin{bmatrix} 1.05 & 0 & -32 \\ 0 & 1.05 & -21 \\ 0 & 0 & 1 \end{bmatrix}$$

Check

We can look at 3 values of x.

1. The far right edge is currently 30, but should be 0
2. The left edge is 1250 right now, but we want it to be 1280
3. Current center is $1220/2 + 30 = 640$, it should stay the same.

$$\begin{aligned} x_{out} &= ax_{in} + by_{in} + c \\ &= ax_{in} + c \\ &= 1.05x_{in} - 32 \end{aligned} \tag{2}$$

$$\begin{aligned} \mathrm{x}_{\text{new left}} &= 1.05(30) - 32 && = -.5 \\ \mathrm{x}_{\text{new right}} &= 1.05(1250) - 32 && = 1280.5 \\ \mathrm{x}_{\text{center}} &= 1.05(640) - 32 && = 640 \end{aligned}$$

Etc

Fonts are weird

```
xdpyinfo | grep dots
```