Tutorial

A2 is out, it's called Inpainting
Tutorial

A2 is out, its called Inpainting
Tutorial

A2 is out, its called Inpainting

How do you think this can be done?
Due to popular demand, there’s no new GUI to develop!

Helper code has been compiled in:
- Visual Studio 2008
- MacOS
- Linux (on CDF)

You’ll read and understand an actual Computer Science paper!
Inpainting

From the target region (region to inpaint)
Inpainting

Choose an initial patch: the ordering is crucial!
Inpainting: Determine a region to inpaint

Find a good source region for the target patch
Inpainting: Determine a region to inpaint

Paste the patch
Go to the assignment description on-line
The Object image computed in the second run contains a lot of pixels that should really be part of the background, but aren't. Equivalently, even though the Alpha Matte should be zero at those pixels, the algorithm assigned non-zero values.
A1 Part B
A1 Part B
A1 Part B
\[
\begin{bmatrix}
R_\Delta & -R_K \\
G_\Delta & -G_K \\
B_\Delta & -B_K \\
R_\Delta' & -R_K' \\
G_\Delta' & -G_K' \\
B_\Delta' & -B_K'
\end{bmatrix}
= \begin{bmatrix}
1 & 0 & -R_K & R_0 \\
0 & 1 & -G_K & G_0 \\
0 & 0 & -B_K & B_0 \\
0 & 0 & -B_K' & \alpha_0
\end{bmatrix}
\]

\[C_\Delta = C - C_K\]
Conversely, there are many pixels that *should be fully opaque*, but aren't....
A1 Part B
A1 Part B
A1 Part B Question 2

Background 1: (136, 132, 121)
Background 2: (143, 57, 58)
Composite 1: (181, 200, 198)
Composite 2: (185, 193, 178)

\[
\begin{bmatrix}
45 \\
68 \\
77 \\
42 \\
136 \\
120
\end{bmatrix}
= 
\begin{bmatrix}
1 & -136 \\
1 & -132 \\
1 & -121 \\
1 & -143 \\
1 & -57 \\
1 & -58
\end{bmatrix}
\begin{bmatrix}
r \\
g \\
b \\
alpha
\end{bmatrix}
\]

\[
C_\Delta = C - C_k
\]
A1 Part B Question 3
A1 Part B Question 3
A1 Part B Question 3