

for SoundHound

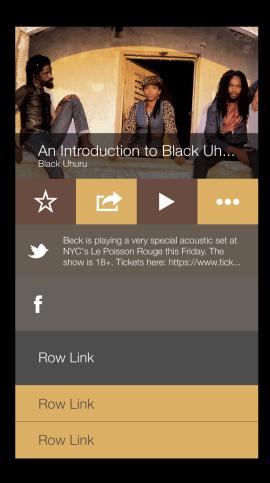
Why?

Extracting colors from subject images creates customization and variation.

- The app will feel tailored to each user's experience.
 - "This screen looks like it was made just for my favorite album!"
- Changing color schemes will prevent the app from feeling monotonous (literally).

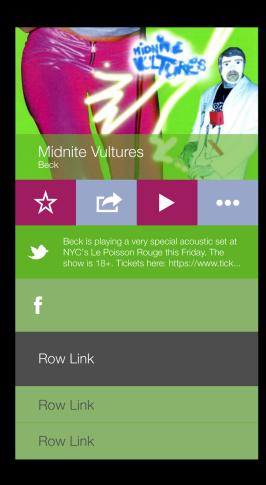
Risks

- Dilution of identity:
 - Will changing colors impact the perception of SoundHound's ownership of the experience?
- Ugly screens:
 - Colors should be carefully controlled and used to avoid obnoxious palettes.
- Performance:
 - All sampling must be client side without an image metadata backend.

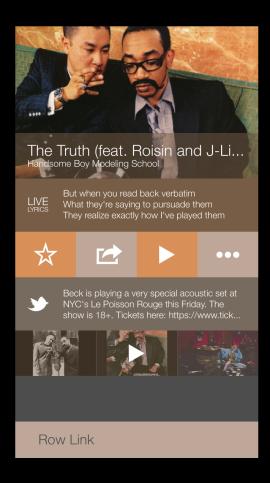








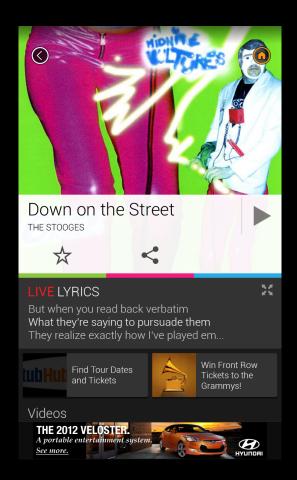








But when you read back verbatim What they're saying to pursuade them They realize exactly how I've played em...





Possible Implementation: Two-Layer



Possible Implementation: Angular Gradient

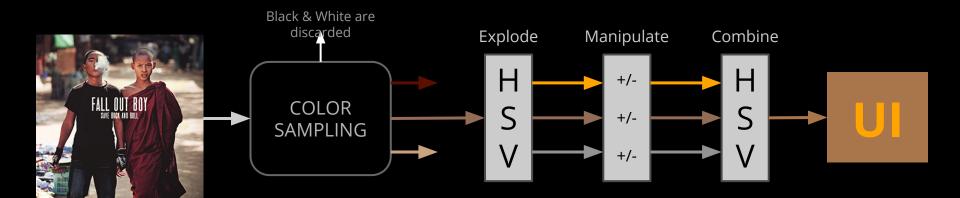


How?

There are many color sampling algorithms out there. We want ones that do more than just average colors.

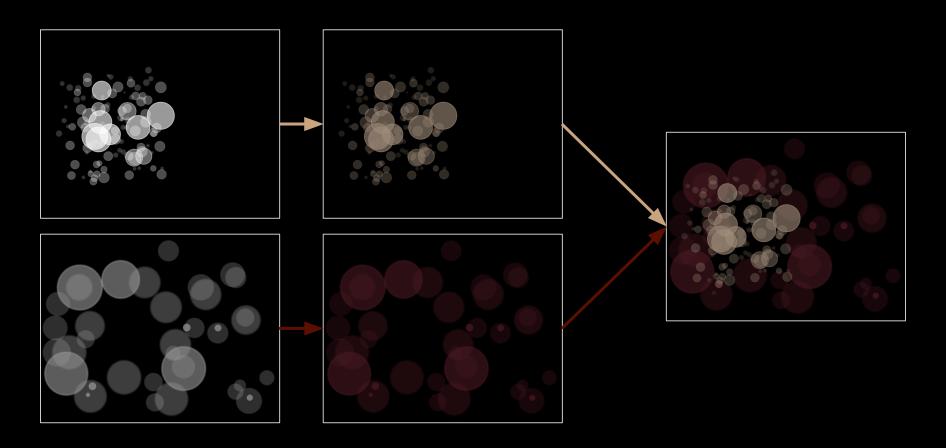
- An ideal algo will extract an arbitrary number of colors ranked by dominance.
- There are hacky ways to do it (GIF conversion, resizing) but they may not be efficient or return the best colors.

How?

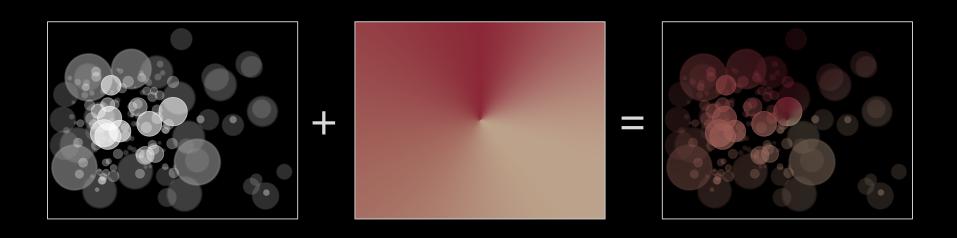




Implementation: Two-Layer



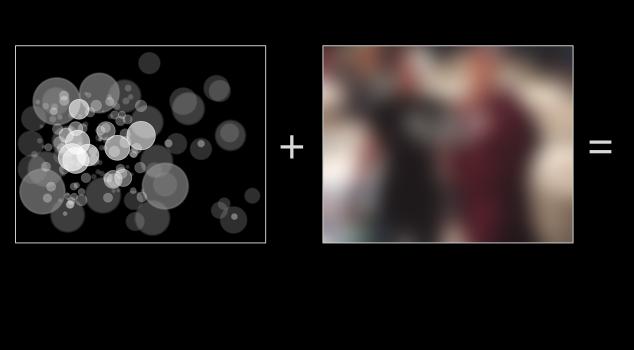
Implementation: Angular Gradient



Hacky: Blurred transparency



Hacky: Blended transparency



Multiply



Overlay

