

Notes - Design - Vision

- Dr Nick Hayward

Basic intro to vision and its consideration relative to application and interface design.

Contents

- Intro
- Peripheral vision
- Foveal resolution
- Is peripheral vision any use?
- Application in user interfaces
- Make messages visible
- Overt interface options
- Positive highlighting and focus
 - design pop
- Resources

Intro Another aspect of human vision that we need to consider for interface design is the nature of spatial resolution.

Peripheral vision As human vision moves from the centre to the edges, the associated spatial resolution will drop greatly. There are three known reasons for this phenomenon:

- Data compression - information from the visual periphery is compressed, with associated data loss as well, before it is transmitted to the brain. The equivalent process at the centre of vision does not experience the same loss of information.
- Pixel density - each human eye has between 6 to 7 million cone cells in the retina. In the centre of our vision, these cone cells are more densely packed, in a region known as the *fovea*, than at the eye's periphery. Effectively, it's almost akin to a focal point in photography which then gradually fades to the edges. Focal point to a background bokeh effect.
- Processing resources - the centre of human vision, the fovea, accounts for only approximately 1% of the retina. However, our brain's visual cortex devotes approximately 50% of its area to input from the fovea. The other half of the visual cortex is used for processing the data from the retina's remaining 99%.

The result of the above is that human vision has far greater resolution at the centre than elsewhere. The disparity in resolution between the centre and periphery is, therefore, pretty stark.

Foveal resolution If we look at this concept for foveal resolution, we see that the vast majority of the resolution is in the centre with an ever decreasing resolution as we move out. It has often been described as the equivalent of looking through a frosted door at the periphery of our vision. The illusion is that we are not as aware of this deficiency because as we move our focal point, our eyes track remarkably well to keep the whole visual world in focus. Our brain also helps by filling in many of the missing pieces in our peripheral vision.

Foveal Image (source: [Illustrated Dictionary of Computer Vision](#))

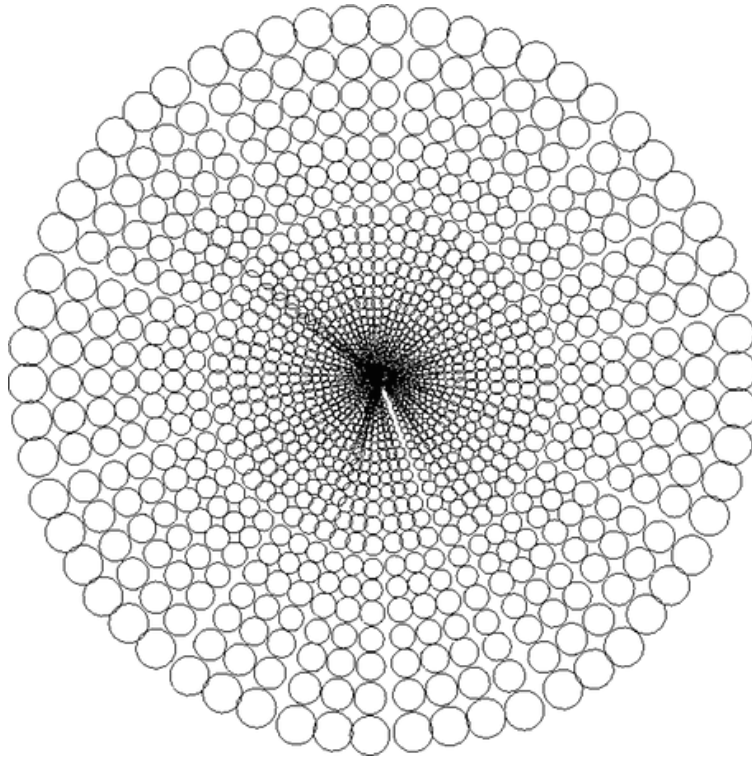


Figure 1: Foveal Image

Is peripheral vision any use? With all of this in mind, we need to consider whether peripheral vision is actually any use.

Our peripheral vision is beneficial because it serves three primary functions,

- it provides better vision in the dark
- helps us detect motion
- and also helps guide the fovea, or the centre of our vision

Application in user interfaces So, how does this apply to user interfaces for computers?

Well, the low resolution and focus of our peripheral vision is often cited as one of the reasons why users, of websites or mobile apps for example, may fail to notice error messages in some aspects of their application or website. When a user clicks on a link or button, it's normally the focal point of their fovea as well.

Therefore, if an error message is presented in the scope of peripheral vision relative to the clicked link, for example, it should not be a surprise if the error message is either ignored or there is a delayed response. Error messages should be obvious relative to the perceived focal point of a user.

Other issues can also affect our peripheral vision for such messages and output. Standard design considerations, such as colour, font or icon size, relative positioning, opacity, and so on, also play a part in whether our user is aware of the error they have committed within our application or website.

Make messages visible If we know there are issues with messages and vision, what can we do to improve their visibility for users.

- use a user's focal point to our advantage - i.e. put the message where the focal point should be. Users tend to focus in a predictable manner when interacting with user interfaces. This inherent predictability can be used to help us as designers.

For example, users, in the west at least, follow a pattern for forms, control panels &c. that moves from upper left to lower right. When a user clicks a link or button, it can usually be assumed that the user is looking directly at it, and for a few brief moments afterwards as well.

- we may also mark an error prominently to help indicate to our users that something requires their immediate attention. This can normally be accomplished by placing the message near the source of the error. However, if this places the message too far from the focal point it will need to be relocated.
- as with colour usage and representation, consider adding an icon or symbol that clearly defines an error. It will also need to be consistent throughout our interface design.
- we may also reserve a single colour for error messages within our interface. This is customarily red, for example Bootstrap's danger alert class. However, colour is always an issue relative to overall design.
- For example, what happens if red clashes with a company's logo or brand image. This would be a good example of where a supporting icon would help reinforce the error message.

Overt interface options There are also more obvious methods available to us as designers in our goal to attract a user's attention. Use with care, however.

- Option 1 - a message in an error dialog or modal box. This option places the error message front and central, right in the user's face thereby making it hard to ignore and miss. However, be aware that a user's annoyance with certain message types can easily escalate as the modal nature can take over the screen and the application. Non-modal allows a user to continue working and ignore such error messages. Therefore, carefully consider context for modal error messages. Also, remember that traditional pop-up messages are often overridden by user settings in modern browsers. If an error message opens in a new tab, it's use will, of course, initially be limited to a user.
- Option 2 - use sound to reinforce an error message. A system audio beep or alert is a common way of audibly notifying a user. A user will then be able to scan the user interface for an accompanying message. The value of an audible beep is that it allows us to display non-obstructive messages within the user interface, and then alert the user to check their screen. However, this is, at best, a supporting option to visual messages. It can quickly become annoying, a user can turn the volume down, the sound becomes in-audible in noisy environments, and so on. It is, however, useful as a supporting option. In mobile and wearable applications, this option has also been supplemented with vibrations.
- Option 3 - animated notifications, such as a brief blink or bounce, works in tandem with our peripheral vision's ability to detect motion. This detection then causes reflexive eye movements, which brings the motion into the fovea. This motion only needs to be momentary, it doesn't take much motion to trigger eye movement towards the screen. It's often considered akin to a person shaking their head as a reinforcement to a verbal 'No' or an indication of something incorrect. This type of notification motion is often seen in user interface menus.

For example, a user selects save, copy, paste, &c. from a menu and the menu briefly blinks to confirm selection.

Naturally, as with popups, motion should be used sparingly in a user-interface. There is also often the association of motion and popups &c. with advertisements. If it's a mobile application, for example, you can also give more experienced or conditioned users the option to turn off such motions and messages. As a designer, the choice is yours.

Positive highlighting and focus As we've already seen, one useful function of peripheral vision is the trigger it provides to the fovea to determine point of focus. Moving objects in our peripheral vision will quickly draw our attention.

The other important role of our vision is in searching for things, and again peripheral vision has a key role to play.

However, this peripheral aid is also dependent upon our search target. Is it hidden in a crowd, standing out with a bright, obvious colour, moving quickly through our field of vision, and so on.

As designers, we can obviously help our user's focus on their search target by providing visual aids. Bold is a very useful tool for designers, it allows a search keyword, result to be highlighted, what is known in design terms as 'pops', short for 'pops out'.

design pop Design *pop* can also be applied to other aspects of interface design. We use it regularly to denote and indicate visual hierarchy within our designs. For example, titles, headings, text boldness, bullet points, indenting, and so on.

We also see it used to indicate status, again normally reserving red, for example, for problems and warnings. For example, our image of Google Maps uses a colour scale to quickly denote traffic density.

Resources

- Krug, S. *Don't make me think, revisited: A common sense approach to web usability*. 3rd Edition. New Riders. 2014.
- Norman, D. *The Design of Everyday Things*. Basic Books. 2013.