Written by Marco Attard 22 February 2012

As seen in films like *Robocop* and *The Terminator*, science fiction predicts an era of augmented reality (AR) lenses streaming data directly into our eyeballs. Now Google is working to turn such fiction into reality, at least according to the New York Times.



The report quotes unnamed Google sources saying the company will start selling high tech glasses with a form of heads-up display

by the end of this year. Expected to cost "around the price of current smartphones" (anywhere from \$250 to \$600) the glasses will supposedly be Android-powered, with a separate mini-display sitting "a few inches" from the wearer's eye.

The glasses should also have 3G or 4G internet connections, motion sensors and GPS, while a low-resolution built-in camera monitors the world and overlays the display with information on the area in real-time.

Interestingly, the *NY Times* says some top people are working on the project, deep inside the secretive Google X offices-- including co-founder Sergei Brin as "key leader" and Latitude creator Steve Lee. One *NY Times* source says the glasses will use modified versions of current Google products, such as Google Maps.

The blog *9 to 5 Google*, which reported earlier this year on the "Goggles" project, gives some other details on the navigation system (using head tilting to scroll and click) and the glasses' actual look (similar to <u>Oakley' Thump sunglasses</u>).

AR Glasses: From Sci-Fi to Sci-Fact?

Written by Marco Attard 22 February 2012

Will customers look up to the wearable form factor? Insiders say the "Google goggles" project is experimental, not aiming to sell large volumes. But who knows, maybe the 21st century is yet to become the era of the wearable computer (on our wrists AND faces)... and start looking more like the future that sci-fi promises. Now, where are our jetpacks and flying cars?

Go Google to Sell Heads-Up Display Glasses by Year's End (The NY Times)

Go HUD Google Glasses are Real and They Are coming Soon (9 to 5 Googl e)