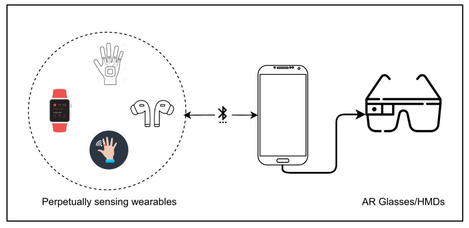
Access Control for Augmented Reality Systems

Sanket Goutam*, **Yoonsang Kim***, Amir Rahmati, Arie Kaufman
Ethos Lab, CVC Lab

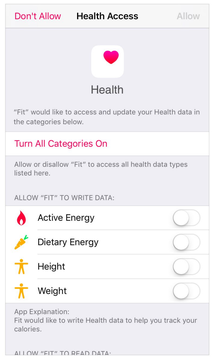
Stony Brook University
Computer Science

Ecosystem of AR Wearables

- Companies like Apple and Google are working towards an ecosystem of AR wearable for immersive user experience.
- Recent prototypes and patent applications suggest two forms of emerging AR ecosystems:
  - Tethered HMDs: uses Smartphone sensors for AR functions.
  - Wearable ecosystem: AR functionality informed by other wearables, like smartwatches.



Existing Access Control Is Insufficient



- Permission control used by developers is analogous to manifest-based model in Android.
- Users typically have a binary gatekeeping mechanism for sensor data access.
- Prior studies showcase ineffectiveness of manifest model.
- New ecosystems, with perpetual access to sensitive sensors, pose threats of surveillance to their users.

AR functionality	# of apps (out of 45)	Sensor Access requested	Functional description
Raycasts	31	Camera	Enables users to place a virtual character or measure distance
Plane Detection	25	Camera	Detects horizontal or vertical flat surface
Face Tracking	3	Camera	Detects a human face
Object Detection	8	Camera	Detects objects
Location Tracking	9	Location	Tracks the location

TABLE 1: Survey of 45 motivate how AR apps remain over-privileged for their sensor access

Reimagining the Permissions Model

Perpetual sensing devices need a permissions model that minimizes over-privilege. The goal of a developer should be to ensure minimal exposure of sensor information to an application, approaching least-privilege, with access that reflect their function use cases.

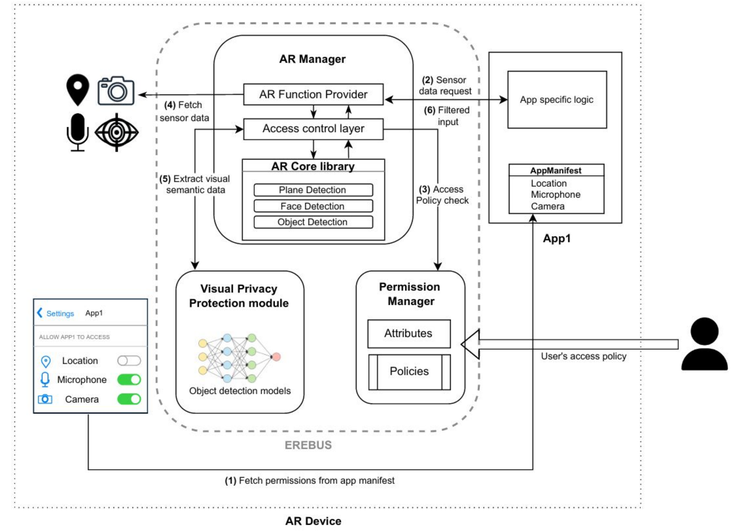
Goals of this framework:

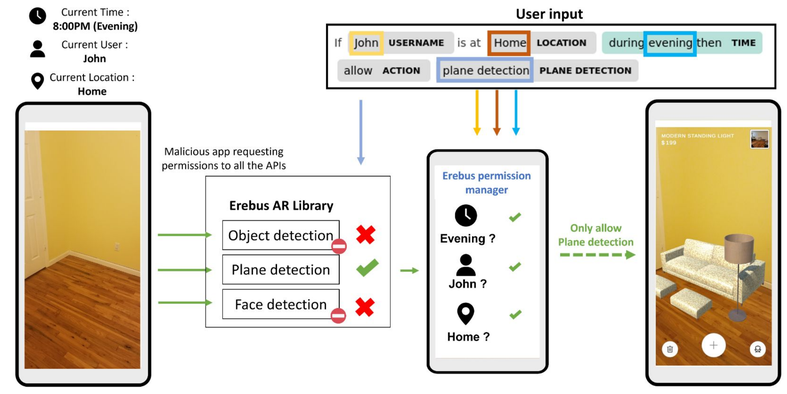
G1 - Regulate direct access to sensors
G2 - Minimize access privileges, at both data and function level
G3 - Data usage transparency to users

Erebus : A Language-based Data Minimization Framework

```
function GetPlane()
{
    let curLoc = GetCurrentLocation();
    let trustedLoc = GetTrustedLocation("Home");
    let curTime = GetCurrentTime();
    let validHour = GetValidHour("Evening");
    let curFace = GetCurrentFaceId();
    let trustedFaces = GetTrustedFaceId("John");
    if ( curLoc.within(trustedLoc) )
    {
        if ( curTime.within(validHour) )
        {
            if ( curFace.matches(trustedFaces) )
            {
                Allow;
            }
        }
    }
}
```

- We introduce a novel domain-specific language specifically designed for AR wearable ecosystem.
- This language allows developers to express precise access control policies over sensor data.
- We developed a framework focusing on visual privacy using ARCore library to demonstrate how it can be adopted by developers.





Takeaways

- ➔ Augmented Reality systems pose new kinds of privacy challenges to uninformed users.
- ➔ Existing Access Control mechanisms are inept for this ecosystem of perpetually sensing devices.
- ➔ We propose a policy framework that enforces language-based data minimization to achieve least-privilege.
- ➔ We migrate 5 existing AR applications to our framework to showcase effectiveness and adaptability.

* Equal contribution