March 19, 2019

PRE-DETECTION GESTURES FOR DETERMINING AN EXPERIENCE FOR A MACHINE-READABLE LINK

HP INC

Pre-detection gestures for determining an experience for a machine-readable link

Scanning machine-readable links are available in two varieties today:

1. ID based experiences: A user points at a physical manifestation of a machine-readable link(QR code, digitally watermarked image etc.) using a scanning device(Mobile phone, hand-held device etc.). The scanning device recognizes the unique image characteristics of the image(hereafter referred to as image recognition) or reads the unique ID or URL that is embedded inside the link(hereafter referred to as deterministic ID recognition) and typically uses a web service(hereafter referred to as the resolver) to fetch an experience that is stored on the web. In addition to the unique ID, the resolver also accepts scan-time parameters that include:
    a. The operating system of the scanning device
    b. The physical location coordinates of the scanning device

    These inputs are used by the resolver to personalize or customize the experience sent back to the scanning device. All these inputs passed to the resolver are existing art.
2. Augmented reality(AR) based experiences: A user points a scanning device at a physical manifestation of a machine-readable link. The scanning device recognizes the link using image recognition or deterministic ID recognition. An augmented reality experience is then shown where the reference object is kept in the viewfinder of the scanning device and relevant experience is played. A simple example is https://www.youtube.com/watch?v=Runmu9JEcH0. In the case of AR, the boundaries of the link are used to anchor the experience to these bounds. In the example given, the boundaries of the business card are used to anchor the 3D animation. This is also an existing art.

Novelty and use-cases:
This disclosure proposes a novel approach where gestures that are captured just **before** the link is detected(either through image recognition or deterministic ID recognition) are passed as input to the resolver before retrieving the experience. This enables the resolver to consider the pre-detection gestures in order to select the right experience to be returned to the scanning device.

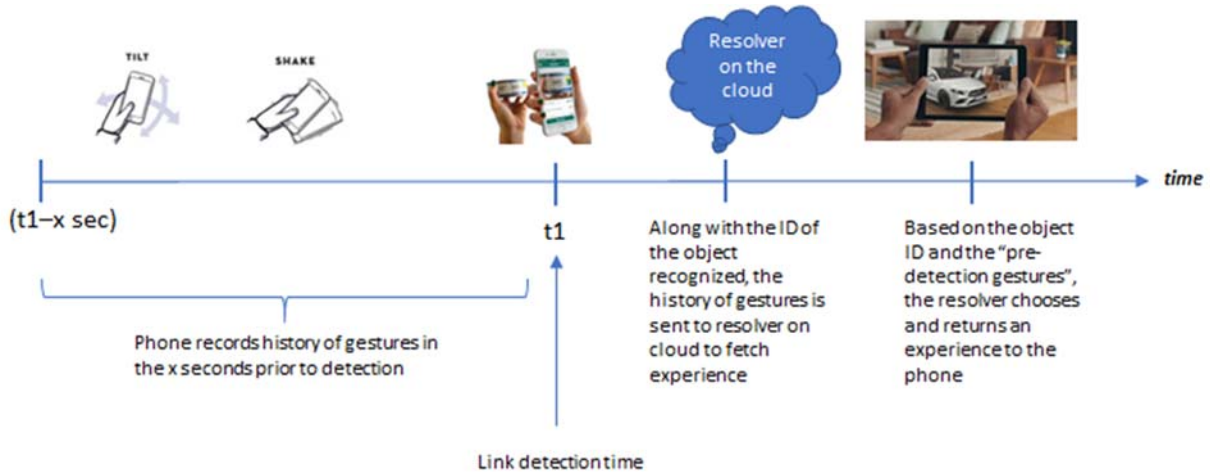


Fig 1: Timeline of predetection and experience rendering

In order to achieve this(refer to the Fig 1),

- The scanning device always maintains a short timeline of pre-detection gestures. As an example, this could be for a duration of ten seconds - but the duration can be customized by the experienced author. This means that the scanning device will have a record of the gestures that have occurred in the last ten seconds, but anything older than that is constantly purged.
- Gestures could be actions that are captured using the phone sensors(gyroscope, accelerometer etc.) that could include hand-twisting actions, hand-chopping actions, patterns drawn in the air with the movement of the phone etc. They could also be captured as directional movements of the phone from right to left or up to down.
- On link detection, the pre-detection gesture timeline of the last few seconds is sent to the resolver. The resolver uses this data to choose between multiple available experiences or to just customize an experience.

Use-cases

Some interesting use-cases where this could be applied:

1. Gaming: If the pre-detection gesture included a bottom to top movement of the phone, the experience that has a 3D aircraft animation will show the aircraft taking off. If the pre-detection gesture involved a top to bottom movement of the phone, the experience will show the aircraft landing.
2. Security/Gaming: If the pre-detection gesture included a specific sequence of gestures, then a secret/premium experience would be unlocked. Example: A twist of the phone followed by a chopping action of the phone which is followed by the detection of the link would show a secret experience. The pattern can be pre-configured as the “unlock gesture sequence” by the experienced author.
3. Personalization: Let us imagine that the experience that is shown on scanning the link is one for folding and creating paper art(Origami). Based on the interpretation of the gesture, the system could make the interpretation of whether the user is left-handed or right-handed. Using this interpretation, the resolver could make the choice to show either a left-handed or right-handed person in a video of a person folding paper art.

System diagram

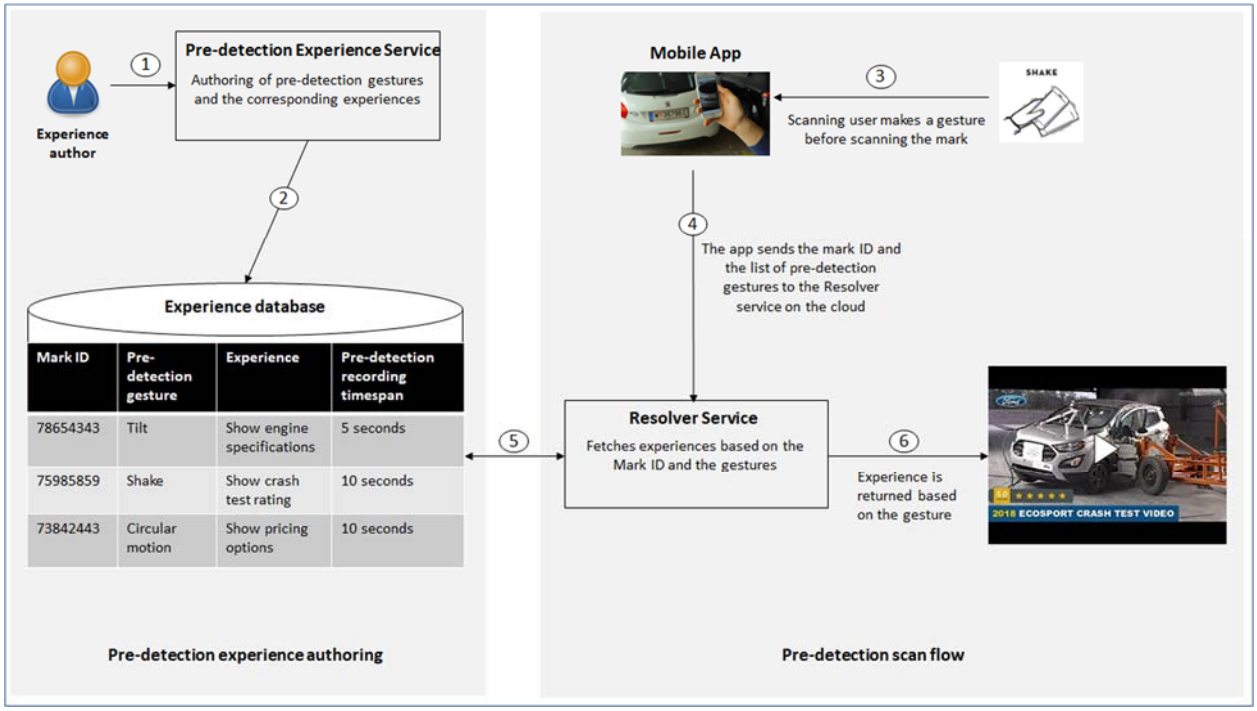


Fig 2: System flow

The overall flow can be separated into two sequences with steps as numbered in Fig 2:

Pre-detection experience authoring

1. The system will provide a tool (User interface or API) for authoring pre-detection based experiences. Here, the experience author will specify the Mark ID for which the experience is being authored, the gesture(or sequence of gestures) that needs to be registered before the detection of the mark and the experience that needs to be presented on fulfilling these criteria. Additionally, the experience author can also specify the number of seconds prior to detection that needs to be considered by the resolver service when the scan is being processed.
2. Once authored, these instructions will be stored in a database on the cloud as described in Fig 2.

Pre-detection scan flow

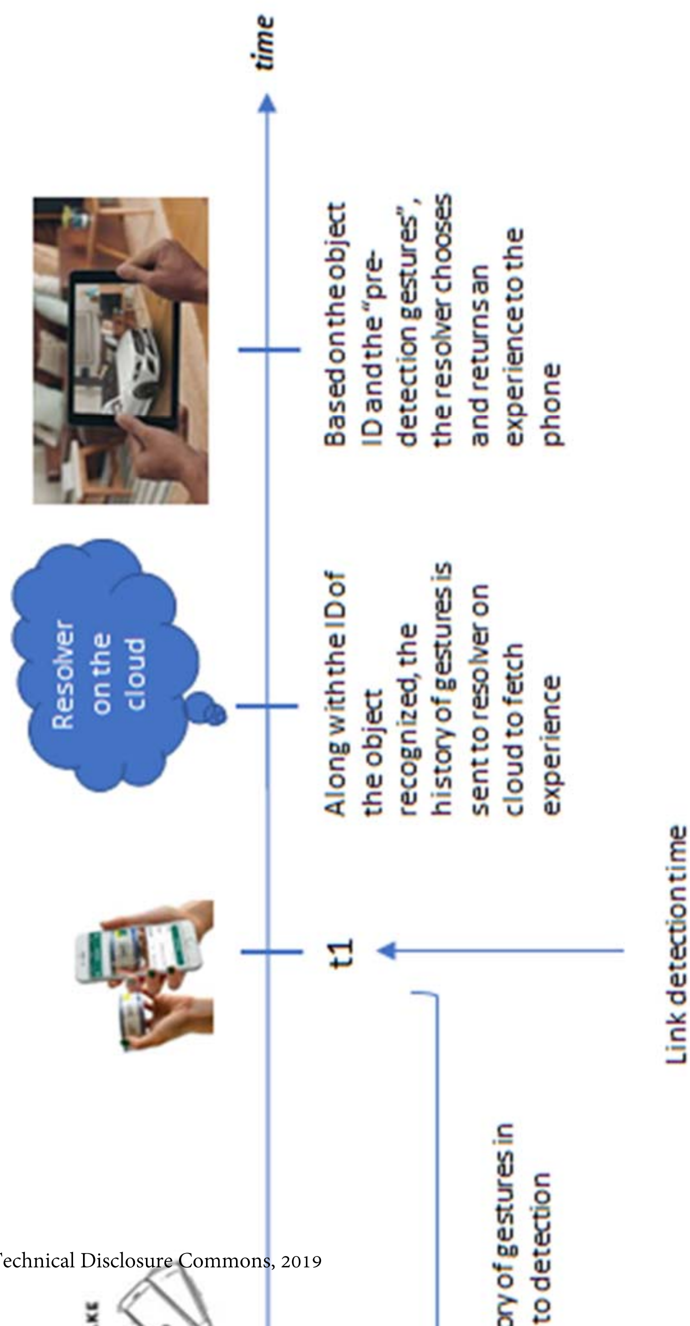
3. The scanning user opens the scanning app on a smart-phone and performs a gesture(or a sequence of gestures) prior to focusing on a mark.
4. The scanning app also maintains a continuous record of gestures made in the last ‘x’ seconds(purging older data continuously). The value of ‘x’ can be customized in the app. The scanning app detects the machine readable code on the mark and on detection, sends data about the unique ID of the mark and the gesture data captured in the last ‘x’ seconds to the Resolver service.

5. The resolver service combines the data sent by the scanning app and matches it with the data in the experience database(as shown in fig 2) and chooses the experience that needs to be returned to the scanning app.
6. The experience is sent back to the scanning app and the experience is shown to the scanning user.

Additional extensions:

- A single gesture could be split into multiple actions. Ex: A twist of the phone between 11 deg and 46 deg will lead to Experience-1 whereas the same twist gesture between 47 deg and 89 deg Experience-2.
- The experience could be chosen based on a sequence of gestures with a specific order (described in the use-cases section).
- The difference in intensity of the gesture could result in a different experience being returned to the scanning app. This is especially interesting in gaming use-cases where the intensity of the gesture indicates a characteristic of severity that can return an experience suited for that situation.

Disclosed by Priyank Agrawal, Sidhant Das, Udita Jindal, Sandeep Prabhu and Pulkit Tanwar, HP Inc.


time
Based on the object ID and the "pre-detection gestures", the resolver chooses and returns an experience to the phone
Resolver on the cloud
Along with the ID of the object recognized, the history of gestures is sent to resolver on cloud to fetch experience
t1
Link detection time