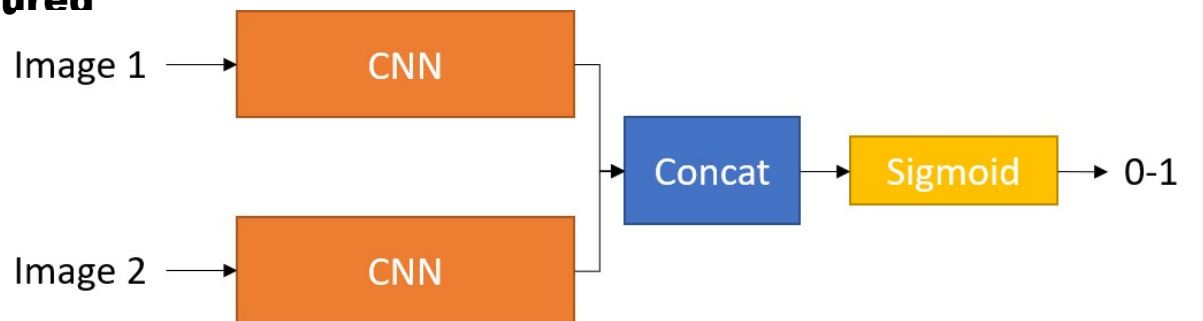


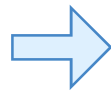
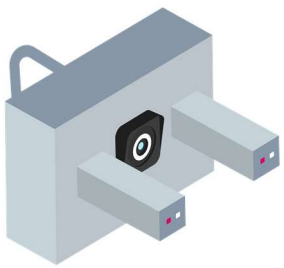
How MANY TIMES have you gone to your favorite retail store and didn't **MAKE** a purchase because of the long checkout queue? As a store owner, aren't you bothered by inefficiencies in **MANUAL** billing, absorbing **TIME** that **CUSTOMER** could've used to purchase **MORE ITEMS**? Shopping carts have been around for ages, with no innovation, we at **OptiSenseCart**, integrate **AI** with heterogeneous **COMPUTING** to provide **SEAMLESS** shopping experience, keeping **CUSTOMER** satisfaction as our **PRIME** focus. Using **AI**, we recognize the products being placed in the cart and create a cashier-less, **CUSTOMER-CENTRIC** and hassle-free shopping. This results in increased footfalls, lower resources, and increased efficiency. Now, who would want to return to a store that has the **SAME** old cart again? **Digitize with OptiSenseCart.**

Algorithm

OptiSenseCart Recognizes products captured by the **On-Cart Module** by using a **SIAMESE Network** which **SEMANTICALLY COMPARES** its feature vector with the feature vectors of the product dataset and accordingly ranks the products **FROM** the dataset based on **SIMILARITY**.



Implementation

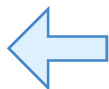
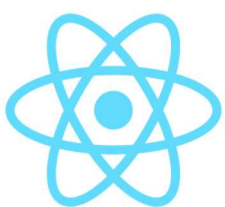


The **On-Cart Module** consists of two **Infrared Sensors** and a **CAMERA**. The sensors help understand if the product is being added or **REMOVED** FROM the cart. When the sensors get triggered, it captures an **IMAGE** and **POSTs** it to the **Inference API**

The **Inference API** is hosted on **Google Cloud Functions** which receives the **IMAGE** and uses our **MODEL** for **SEMANTIC COMPARISON** of the query **IMAGE** with the **IMAGES** in the product set. The result are then updated in **Firestore REAL-TIME Database**.



Firestore Realtime Database is where the **User's Cart and Sessions** are stored and also acts as the **serverless backend** for **SenseCart's Web App**.



OptiSenseCart's React App allows the **CUSTOMER** to start a cashier-less shopping session with just a **QR code scan** on the cart and keep track of the cart and **MAKE PAYMENT**.

Results and Discussion

