Mini Info-pedia

This is a personal information center meant to help explain some of the terms that I use - from my point of view.

Development Board

A **development board** is meant to help system developers to learn the inner workings, interfaces and features of a certain processor. Thus, it usually contains minimal components that are required to start up a system based on that certain processor. However, many development boards are currently equipped with features comparable to that of a desktop computer's motherboard. This allows them to be used in virtually any test scenarios due to the fact that they provide basic input and output interfaces that are easily available off-the-shelf.

Many will be more familiar with **microprocessor development board** instead, but since we have other processing units such as Digital Signal Processors (DSP) and Field Programmable Gate Arrays (FPGA) that are also available on boards with similar functions, it would be more appropriate to generalize the term.

With the ever-increasing features of development boards, many scientific research have started to use them as a system platform.

All that been said, development boards are typically intended for prototyping and not usually used in an end product.

Read more @ wikipedia

Embedded System

An **embedded system** is a combination of hardware and software (sometimes the term *firmware* is used as well) components that is usually meant to be part of a larger system (thus the word *embedded*). However, it is technically a standalone system (one that works without any external support) and is primarily designed for a specific purpose (i.e. with predefined tasks and dedicated functions). With this in mind, an embedded system requires minimal user input (minimal user interface, not re-programmable by end-users) and provides only the expected output depending on its application.

Since an embedded system only needs to do a specific task, it can most of the time respond to real-time events (system input). The occurrence of these 'real-time' events can even be predicted due to the fact that a predefined task usually have a known set of inputs. Based on this scenario, an embedded system does not (most of the time) require a state-of-the-art central processing unit (CPU) to operate. Consequently, the preferred CPU for an embedded system is a microcontroller, which is basically a microprocessor that emphasizes self-sufficiency (e.g. on-chip memory, minimal glue logic) and therefore offers a cost effective solution.

Current implementations of embedded systems are made programmable so that the main system

developers can easily make appropriate upgrades or modifications. This enables modern embedded systems to have more features (upgradable at any time) at a lower cost (none or minimal hardware changes).

Read more @ wikipedia

Computer Vision

Research work related to vision comprises many different field of studies - each of which have different point of view on how it can be achieved and what feature should be emphasized. Therefore there a few variations to the name of this particular field of study - but they all actually try to achieve the same thing. It is mainly referred to as computer vision in the field of computer science, while the term machine vision (or robot vision) is more popular in the engineering discipline. Another familiar name would be computational vision, where interdisciplinary researchers get together to understand biological vision and create a system that can 'see'.

Computer vision is closely related to image processing and therefore a sufficient understanding of image processing techniques is required. Naturally, it is also the inverse of computer graphics and in fact, a substantial amount of work on vision systems have used graphics accelerator (graphics card) as the processing platform. Computer vision also requires some form of artificial intelligence, which separates it from basic image processing.

As defined by Marr (1982), vision is actually the process of knowing what and where an object is in its environment (something like that.. I'll refine this later). So, the basic tasks pursued by vision is image understanding (what it is), image analysis (where it is) and scene analysis (environment). The full flow can be further divided into 2 levels:

- early vision
  - low-level
  - image processing?
  - feature analysis/extraction, image segmentation
- scene analysis
  - high-level
  - knowledge-based processing
  - shape analysis, object recognition & localization

Main reference:

```
@book{993688,
 author = {Edwin D. Reilly},
 title = {Concise Encyclopedia of Computer Science},
 year = {2004},
 isbn = {0470090952},
 publisher = {John Wiley \& Sons},
 }
```

Read more @ wikipedia

From:
https://azman.my1matrix.cc/ - **Azman @MY1**

Permanent link:
https://azman.my1matrix.cc/doku.php?id=archive:infopedia

Last update: **2026/04/04 13:40**