# 

Mục Lục

[Tổng quan thiết kế dự án Leap Motion Mouse 2](#_Toc7966460)

[1. Tổng quan về dự án 2](#_Toc7966461)

[1.1. Đặt vấn đề 2](#_Toc7966462)

[1.2. Mô tả bài toán từ góc độ người dùng 2](#_Toc7966463)

[1.3. Ứng dụng thực tế của bài toán 2](#_Toc7966464)

[1.4. Sơ đồ khối của hệ thống 3](#_Toc7966465)

[2. Thiết kế hệ thống 3](#_Toc7966466)

[2.1. Phân tích sơ bộ hệ thống 3](#_Toc7966467)

[2.2. Thiết kế xử lý chuyển động bàn tay 4](#_Toc7966468)

[2.3. Driver điều khiển thiết bị 6](#_Toc7966469)

[3. Phân công giải quyết công việc 6](#_Toc7966470)

# Tổng quan thiết kế dự án Leap Motion Mouse

## Tổng quan về dự án

### Đặt vấn đề

Hiện nay công nghệ thông mình đang phát triển rất mạnh mẽ. Chẳng hạn có thể kể đến như tivi điều khiển bằng giọng nói, hay các thiết bị được điều khiển thông qua ứng dụng di động,… Với sự phát triển mạnh mẽ của công nghệ IoT và trí tuệ nhân tạo, việc điều khiển thiết bị ngày càng trở nên đa dạng hơn chứ không gói gọn ở việc điều khiển bằng tay thông thường. Có thể nói công nghệ trí tuệ nhân tạo đã góp phần quan trọng làm cho các thiết bị trở nên thông minh hơn.

Chúng ta có thể nghe nhiều tới thiết bị điều khiển bằng giọng nói. Tuy nhiên, việc điều khiển thiết bị bằng cử chỉ có vẻ khá xa lạ. Bằng cử chỉ của bàn tay ta có thể ứng dụng vào giải quyết rất nhiều vấn đề. Có thể kể tới những lĩnh vực như ứng dụng cho người khuyết tật (ngôn ngữ cử chỉ), điều khiển thiết bị trong nhà thông minh,… Chính vì những ứng dụng này mà đội đã quyết định thực hiện đề tài liên quan đến điều khiển thiết bị thông qua cử chỉ bàn tay với tên gọi Leap Motion Mouse. Mô tả sơ bộ về dự án sẽ trình bày chi tiết ở phần sau.

### Mô tả bài toán từ góc độ người dùng

Ở dự án này, thông qua việc nhận dạng cử chỉ của bàn tay, người dùng có thể điều khiển các thiết bị trong nhà như đèn điện, tivi, điều hòa…

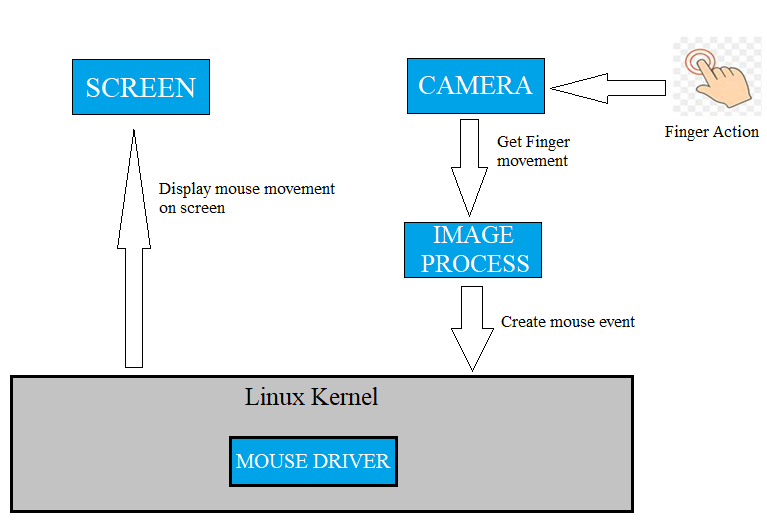
Một ví dụ thực tế: Người dùng đứng cách xa tivi 3m. Trên màn hình tivi có hình một con trỏ chuột. Mỗi khi ngón tay người dùng di chuyển, con trỏ chuột trên màn hình sẽ di chuyển theo. Mỗi khi người dùng nháy đúp hai lần trên ngón tay của mình, con trỏ chuột trên màn hình cũng sẽ click hai lần liên tiếp. Việc này khiến cho người dùng thao tác với màn hình tivi giống như 1 chiếc màn hình cảm ứng, tuy nhiên người dùng không cần chạm ngón tay trực tiếp vào màn hình, thay vào đó người dùng có thể đứng từ xa và làm việc đó.

### Ứng dụng thực tế của bài toán

Hiện nay  các thiết bị thông minh đang hướng đến việc loại bỏ các thiết bị điều khiển. Thay vào đó chúng sử dụng trực tiếp các khả năng có sẵn của con người như giọng nói, ngón tay,... Việc điều khiển thiết bị qua cử động ngón tay hoặc giọng nói sẽ là xu hướng của tương lai.

Ngoài ra hiện nay một số dòng smart tivi đã cho phép điều khiển bằng giọng nói. Tuy nhiên việc điều khiển bằng giọng nói có nhược điểm là không thể xử lý được các thao tác phức tạp. Nếu có thể áp dụng phương pháp điều khiển bằng ngón tay thì sẽ khắc phục được nhược điểm trên.

### Sơ đồ khối của hệ thống



Hệ thống sẽ gồm camera nhận dạng chuyển động của ngón tay, từ đó tạo ra sự kiện của con trỏ chuột , sau đó gửi tới driver của chuột ở tầng nhân. Từ đó điều khiển con trỏ chuột và hiển thị trên màn hình thông qua cử chỉ tay người.

## Thiết kế hệ thống

### Phân tích sơ bộ hệ thống

Như đã trình bày ở mục 1.4, hệ thống có thể được chia làm hai module chính là : Tầng ứng dụng xử lý chuyển động của bàn tay và driver xử lý dữ liệu của tầng ứng dụng gửi xuống tạo thành chuyển động cho chuột máy tính.

Ở module xử lý chuyển động của bàn tay, yêu cầu đặt ra là ứng dụng có thể gửi về cho mouse driver dữ liệu cần thiết như tọa độ trên màn hình, sự kiện click chuột trái, click đúp chuột trái và click chuột phải. Hình ảnh các chuyển động của bàn tay được chụp lại thông qua một camera gắn với board udoo. Mỗi frame được camera chụp lại sẽ tiến hành xử lý và gửi tín hiệu. Ở module này phải đảm bảo thời gian xử lý nhanh, tiêu tốn ít tài nguyên của board udoo để cho người dùng có trải nghiệm tốt nhất.

Module thứ hai là mouse driver xử lý dữ liệu tầng ứng dụng gửi xuống. Ở module này, yêu cầu đặt ra là từ dữ liệu được gửi xuống sẽ tạo thành các chuyển động như một chuột máy tính thông thường có thể làm được đơn giản như di chuyển, click chuột trái, click đúp chuột trái và click chuột phải.

### Thiết kế xử lý chuyển động bàn tay

#### Phương hướng giải quyết

Hiện nay, lĩnh vực thị giác máy tính đang phát triển rất mạnh. Hàng loạt các thuật toán xử lý ảnh tiên tiến đã ra đời. Cùng với đó là việc phát triển mạnh của deep learning đã góp phần rất lớn vào giải quyết bài toán theo dõi và nhận diện chuyển động. Ở bài toán này, để đơn giản thì ta sẽ chia thành hai vấn đề chính là theo dõi (tracking) theo chuyển động của bàn tay và nhận diện chuyển động để tạo ra các sự kiện gửi cho tầng kernel xử lý.

Vấn đề theo dõi chuyển động của bàn tay có nhiều phương hướng giải quyết. Có thể đơn giản sử dụng một số thuật toán xử lý ảnh phổ biến như Color object detect, Background subtraction,… Ta phải đánh giá tính khả thi của các thuật toán để đưa ra lựa chọn thích hợp phù hợp với ứng dụng.

Ở vấn đề còn lại, nhận diện chuyển động của bàn tay ta cũng có nhiều phương án giải quyết. Xử lý ảnh đơn giản ta có thể tìm được bao quanh bàn tay hay còn gọi là contour. Từ đó xử lý và đưa ra những dự đoán về chuyển động như một ngón tay, hai ngón tay hoặc nắm đấm,… Ngoài ra để nâng cao độ chính xác ta có thể sử dụng những giải thuật deep learning phổ biến trong công nghệ nhận dạng ảnh như Convolutional neural networks (CNNs).

#### Phân tích ưu nhược điểm của các phương án

Thứ nhất là việc lựa chọn giữa color object detect và background subtraction. Ở giải thuật Color object detect, ta sẽ theo dõi vật dựa trên màu của vật thể, ở đây là bàn tay người, Bằng việc sử dụng phổ màu HSV ta có thể nhận biết được vật thể đang ở đâu trong ảnh và từ đó xác định được ô bao quanh vật. Tuy nhiên thuật toán sẽ mất đi độ chính xác khi trong ảnh có vật gì đó màu giống da của người, lúc đó việc dựa theo màu sẽ không còn ý nghĩa. Còn ở giải thuật Background subtraction ta sử dụng một ảnh nền và với mỗi frame mới ta tiến hành trừ đi ảnh nền đó để phát hiện ra vật chuyển động. Về cơ bản thuật toán hoạt động khá tốt, chỉ có tồn tại là thuật toán nhạy với ảnh nắng, khi bàn tay che mất ánh nắng thì ánh sáng thay đổi, thuật toán sẽ hoạt động không tốt.

Về giải thuật nhận diện chuyển động bàn tay bằng cách tìm contour và nhận dạng sự thay đổi của contour để đưa ra chuyển động, ta thấy ưu điểm nổi trội đó là đơn giản, chi phí bộ nhớ thấp. Nhưng bù lại độ chính xác không cao bằng sử dụng giải thuật Deep Learning. Ở giải thuật Deep Learning, ta có thể thấy được ưu thế vượt trội ở độ tin cậy với kĩ thuật nhận dạng với độ chính xác cao. Tuy nhiên sẽ tiêu tốn một chút tài nguyên trong việc tính toán. Mặc dù vậy, với công nghệ phần cứng mạnh như hiện nay thì đây có thể không là vấn đề nghiêm trọng.

#### Lựa chọn giải thuật

Sau khi phân tích ưu nhược điểm của từng giải thuật, vì mục đích đơn giản và tối ưu hóa cho ứng dụng nhóm đã lựa chọn giải thuật cho từng vấn đề. Ở phần theo dõi chuyển động bàn tay nhóm sẽ sử dụng giải thuật background subtraction và vấn đề nhận dạng chuyển động sẽ được giải quyết bằng một mạng CNNs.

Ở bài toán theo dõi chuyển động, lựa chọn giải thuật background subtraction là phù hợp. Vì ứng dụng là mô phỏng chuột máy tính, ta sẽ sử dụng phần lớn là ứng dụng trong nhà nên việc ánh sáng ảnh hưởng là ít. Ngoài ra camera theo dõi chuyển động là cố định nên việc ghi lại background là hợp lý.

Việc sử dụng mạng nơ rôn tích chập (CNN) để giải quyết bài toán nhận dạng nhằm mục đích nâng cao độ chính xác. Việc tính toán của phần cứng bây giờ tương đối nhanh nên việc sử dụng một mô hình Deep Learning là hợp lý để nâng cao độ tin cậy.

#### Platform và công cụ giải quyết

Để ứng dụng những giải thuật đã lựa chọn ở trên, cần lựa chọn platform, công cụ được hỗ trợ nhiều hiện nay. Có thể kể đến là bộ thư viện opencv, các framework hỗ trợ Deep Learning như pytorch, keras, tensorflow,.. Để thuận tiện trong việc phát triển thì nhóm sẽ sử dụng Opencv trên nền tảng ngôn ngữ python để giải quyết cả vấn đề đặt ra. Việc các framework Deeplearning được hỗ trợ mạnh trên ngôn ngữ python nên python sẽ là một lựa chọn không tồi. Ngoài ra hiện nay mã nguồn mở cho xử lý ảnh trên nền tảng ngôn ngữ python rất nhiều nên ta có thể được hỗ trợ nhiều trong quá trình phát triển, bảo trì.

Như vậy nhóm đã thống nhất lựa chọn :

* Nền tảng phát triển : Linux.
* Thư viện hỗ trợ : OpenCv.
* Framework hỗ trợ : keras, tensorflow, scikit-learn
* Ngôn ngữ lập trình: Python

### Driver điều khiển thiết bị

Việc phát triển driver điều khiển thiết bị có thể thực hiển bằng cách phát triển một mouse driver( driver điều khiển chuột máy tính). Driver sẽ nhận tín hiệu từ tầng ứng dụng gồm có tọa độ thay đổi của bàn tay và sự kiện đi kèm. Việc xử lý các sự kiện và thay đổi tọa độ có thể sử dụng ngắt hoặc ioctl. Ngoài ra cũng có thể sử dụng xử lý signal.

Driver viết ra phải đảm bảo yêu cầu về các chức năng cơ bản của một driver chuột máy tính (mouse driver). Có thể lựa chọn các phương pháp để đảm bảo tốc độ thực thi và không gây xung đột với driver khác trong hệ điều hành linux.

Như vậy ở phần driver điều khiển thiết bị, cụ thể là mouse driver nhóm sẽ sử dụng :

* Nền tảng hệ điều hành linux
* Viết device driver điều khiển thiết bị
* Ngôn ngữ lập trình sử dụng : C

## Phân công giải quyết công việc

Qua thống nhất, nhận thấy ưu điểm của mỗi thành viên, nhóm đã họp và đưa tới quyết định:

* Module xử lý ảnh sẽ do Cường chịu trách nhiệm, đồng thời Trần Hùng Mạnh có trách nhiệm hỗ trợ về phần cứng phục vụ cho module.
* Trần Quang Huy chịu trách nhiệm module mouse driver.
* Trong quá trình thực hiện, hai nhóm nhỏ trao đổi để giúp nhau cùng hiểu vấn đề mình làm, có thể yêu cầu hỗ trợ lẫn nhau.
* Mentor Phạm Vũ Trung Kiên giúp các thành viên giải đáp thắc mắc, giám sát tiến trình thực hiện dự án theo kế hoạch, cung cấp tài liệu cho các thành viên tham khảo.