



دانشگاه تهران
دانشکده‌ی مهندسی برق و کامپیوتر



طراحی مفهومی پروژه سیستم‌های سایبر-فیزیکی پیاده سازی گام شمار با قابلیت تشخیص مسیر حرکت

اعضای گروه:

نسترن علی پور ۸۱۰۱۹۶۵۱۵

ایمان مرادی ۸۱۰۱۹۶۵۶۰

رامین فریاد هریس ۸۱۰۱۹۵۴۴۷

پارسا صدری سینکی ۸۱۰۱۹۵۵۲۶

اساتید:

دکتر مهدی کارگهی

دکتر مهدی مدرسی

۱۳۹۹-۱۴۰۰

فهرست مطالب

۱	مقدمه	۳
۲	کارکرد	۳
۳	ورودی ها	۳
۴	پردازش	۳
۵	شبکه و انتقال اطلاعات	۳
۶	خروجی ها	۴
۷	ملاحظات محیط فیزیکی	۴
۸	نوع حساسیت به زمان	۴
۹	حجم داده ی مورد تعامل	۴
۱۰	الگوریتم های مورد استفاده	۵
۱۱	مقالات و مراجع مورد استفاده	۶
۱۲	تجهیزات و ابزار های مورد نیاز	۶
۱۳	برآورد هزینه	۶

۱. مقدمه

این پروژه یک برنامه اندروید با قابلیت شمارش گام و رسم مسیر حرکت کاربر بدون استفاده از GPS میباشد.

۲. کارکرد

به طور کلی محصول این پروژه یک برنامه برای سیستم عامل اندروید است که شامل سه بخش اصلی گام شمار، تشخیص جهت حرکت و نمایش مسیر حرکت می شود. به علت میزان خطا بالا، مکان یابی تنها با استفاده از انتگرال دوم شتاب های وارد شده به گوشی همراه قابل محاسبه نیست و باید توسط تکنیک های اضافه ای خطا سنسور ها را محدود کنیم. روش پیشنهادی برای حل این مسئله در این پروژه یکی میانگین گرفتن بر روی داده های ورودی سنسور ها با استفاده از تکنیکی مثل فیلتر کالمن هست و روش دوم استفاده از گام شمار برای تشخیص حرکت است.

۳. ورودی ها

- ۱- برای تشخیص جهت حرکت از دو سنسور شتاب سنج و مغناطیس سنج استفاده می کنیم تا جهت تلفن همراه را نسبت به جهت های جغرافیایی بدست بیاوریم.
- ۲- برای گام شمار از سنسور شتاب سنج و API تشخیص فعالیت (Activity Recognition API) خدمات گوگل پلی (Google Play Services) استفاده میکنیم.
- ۳- برای تشخیص این که گوشی در جیب قرار دارد یا دست فرد از سنسور های مجاورت، نور و جاذبه استفاده میکنیم.
- ۴- از سنسور های تشخیص گام و Linear Accelerometer نیز ممکن است برای بررسی مجدد دقت محاسبات و خروجی سنسور ها استفاده شود.

۴. پردازش

تمامی پردازش مورد نیاز برای این برنامه بر روی تلفن همراه صورت می گیرد.

۵. شبکه و انتقال اطلاعات

برای دیباگ و اجرا Android Debug Bridge را بر روی تلفن همراه فعال می کنیم و آن را به کامپیوتر متصل می کنیم و توسط Android Studio کد را کامپایل می کنیم.

۶. خروجی ها

خروجی اصلی برنامه نمایش نقشه مسیر طی شده کاربر به صورت بی درنگ بر روی صفحه تلفن همراه با در نظر گرفتن مقیاس واقعی حرکت و جهت های جغرافیایی هست. علاوه بر خروجی اصلی ذکر شده، برنامه تعداد گام، سرعت حرکت، مقدار کالری مصرف شده، مسافت طی شده کاربر را نیز به عنوان نوتیفیکیشن در هر لحظه به کاربر نمایش می دهد.

۷. ملاحظات محیط فیزیکی

جابجایی خیلی سریع مثل دویدن یا قدم زدن بسیار آهسته ممکن است خطای زیادی در شمارش گام ایجاد کند و امکان محاسبه مسیر حرکت را از بین ببرد. سنسور ها به خصوص سنسور مغناطیس سنج به علت وجود موادی که در فضا های بسته در میدان مغناطیسی تداخل ایجاد می کنند خطاهایی در محاسبه جهت حرکت ایجاد می کنند. حرکت کاربر در هنگام استفاده از برنامه به اشکال مختلف می تواند باشد. مثلاً هنگامی که تلفن همراه در دست کاربر هست و به صفحه نمایش نگاه می کند و قدم می زند، یا اینکه تلفن همراه در جیب کاربر قرار داشته باشد و کاربر در حال حرکت باشد. راه حل این مشکل این هست که نحوه حرکت کاربر را به یک حالت محدود کنیم تا در فاز بعدی پروژه ابتدا نوع حرکت را تشخیص بدهیم و از الگوریتم های مناسب استفاده کنیم.

۸. نوع حساسیت به زمان

بدلیل اینکه در لحظه ثبت هر قدم باید جهت حرکت نسبت به جهت های جغرافیایی را بدست بیاوریم امکان تاخیر بسیار پایی برای محاسبات مربوطه وجود دارد. البته این حساسیت به زمان را تا حدی با میانگین گیری در طول زمان می توان برطرف کرد. با توجه به موارد بالا حداقل سرعت و تعداد هسته پردازشی مورد نیاز برای اجرا برنامه را محاسبه می کنیم و همچنین نرخ بهینه نمونه برداری از سنسور ها را برای کارکرد مناسب برنامه بدست می آوریم.

۹. حجم داده ی مورد تعامل

از آن جایی که سرعت حرکت سریع تر از متوسط برای پیاده روی حدود ۶ کیلومتر بر ساعت هست که حدوداً می شود ۰٫۶ ثانیه و با توجه به قضیه نمونه برداری نایکوئیست-شانون، حداقل فرکانس نمونه برداری باید بزرگتر از دو برابر فرکانس بیشینه سیگنال اصلی باشد که حدوداً می شود ۰٫۳ ثانیه. برای ایجاد مقداری بازه اطمینان برای پیاده روی سریع تر فرکانس نمونه برداری از سنسور ها رو ۰٫۲۵ ثانیه یا ۲۵۰ میلی ثانیه در نظر میگیریم.

برای اجرا با فرکانس ذکر شده حداقل به دو سنسور شتاب سنج و مغناطیس سنج نیاز هست. با توجه به این که هر کدام از این سنسور ها در هر بار نمونه برداری سه داده از نوع float به ما برمی گردانند حجم داده های مورد استفاده ما حداقل ۹۶ Byte/Sec خواهد بود.

۱۰. الگوریتم های مورد استفاده

برنامه از دو طریق می تواند جابجایی را بدست بیاورد، یکی از طریق گرفتن انتگرال مرتبه دوم شتاب می توان مقدار حرکت را بدست بیاورد و یک راه هم این است که از طریق تشخیص هر گام ها و جهت آن گام. روش اول خطا بسیار زیادی دارد و روش دوم هم با یک بار نمونه برداری از سنسور ها مثلا برای تشخیص جهت ممکن است خطا قابل توجهی داشته باشیم. این مشکلات میتوانند با ترکیب کردن سنسور ها یا محاسبه دو بار جواب از دو راه مختلف حل شوند. روش های مختلفی برای این کار وجود دارد که دو مورد آن به اختصار اینجا توضیح داده شده اند.

Kalman filter: که به عنوان تخمین خطی مرتبه دوم نیز از آن یاد می شود، الگوریتمی است که حالت یک سیستم پویا را با استفاده از مجموعه ای از اندازه گیری های شامل خطا در طول زمان برآورد می کند. این فیلتر معمولا تخمین دقیق تری را نسبت به تخمین بر مبنای یک اندازه گیری واحد را بر مبنای استنباط بیزی و تخمین توزیع احتمال مشترکی از یک متغیر تصادفی در یک مقطع زمانی ارائه می کند.

این الگوریتم در دو گام اجرا میشود. در گام پیش بینی، فیلتر کالمان تخمینی از وضعیت فعلی متغیرها را در شرایط عدم قطعیت ارائه می کند. زمانی که نتیجه اندازه گیری بعدی بدست آید، تخمین قبلی با میانگین وزن دار آپدیت می شود. به این ترتیب که وزن اطلاعاتی که دارای قطعیت بیشتری هستند، بیشتر خواهد بود. الگوریتم بازگشتی می باشد و با استفاده از ورودی های جدید و حالات محاسبه شده قبلی به صورت بی درنگ اجرا می شود.

Complementary filter: که ترکیبی از چند فیلتر است و اطلاعاتی را که از ورودی های مختلف دریافت می کند با هم

ترکیب کرده و بهترین نتیجه را به عنوان خروجی میدهد.

الگوریتم تشخیص گام: با استفاده از میزان تغییرات اندازه شتاب بیش از اندازه تعریف شده، یک گام ثبت می شود و برای کاهش خطا شمارش گام هنگامی که کاربر در حال قدم زدن نیست با استفاده از Activity Recognition API موجود در Google Play Services زمانی که کاربر در حال قدم زدن هست تشخیص داده می شود و فقط شمارش گام زمانی محاسبه می شود که کاربر در حال قدم زدن باشد.

تشخیص موقعیت قرار گیری تلفن همراه: با استفاده از سنسور های مجاورت، نور و گرانش زاویه گوشی با افق، میزان نور در محیط (تشخیص تاریکی جیب) و در درست قرار داشتن گوشی محاسبه می شود تا موقعیت گوشی در جیب یا دست تشخیص داده شود.

سودو کد کلی سیستم:

```

while IsWalking() == true do
    if DetectStep(accelerometerData) == true then
        direction = GetDirection(magnetometerData, accelerometerData);
        location = GetLastLocation();
        newLocation = CalculateMovement(location, direction);
        DrawMovementGraph(movementGraph, newLocation);
    else
    end

```

۱۱. مقالات و مراجع مورد استفاده

- https://developer.android.com/guide/topics/sensors/sensors_motion
- https://developer.android.com/guide/topics/sensors/sensors_overview
- <https://www.youtube.com/watch?v=C7JQ7Rpwn2k>
- https://www.researchgate.net/publication/221568690_Pedestrian_Localisation_for_Indoor_Environments
- https://www.researchgate.net/publication/266657734_I_am_a_smartphone_and_I_can_tell_my_user's_walking_direction

۱۲. تجهیزات و ابزار های مورد نیاز

از نظر سخت افزار این پروژه به یک دستگاه تلفن هوشمند مجهز به سیستم عامل اندروید نیاز دارد که حداقل دو سنسور شتاب سنج و مغناطیس سنج را داشته باشد. برای توسعه ی نرم افزار به برنامه Android Studio نیاز است.

۱۳. برآورد هزینه

این پروژه به غیر از یک دستگاه تلفن هوشمند مجهز به سیستم عامل اندروید نیازمندی و هزینه دیگری ندارد.