

# Smartphone: Uma alternativa acessível para aquisição de dados em VANTs

**Leonardo Mariga**

**Rafael Araújo Lehmkuhl;**

**Prof. Amir Antonio Martins Oliveira Jr., Ph.D.**

UFSC - Universidade Federal de Santa Catarina

## RESUMO

Os *smartphones* possuem uma grande variedade de sensores, como GPS, giroscópio, acelerômetro, magnetômetro e barômetro, que são utilizados no dia-a-dia para orientação, localização e conforto de seu usuário. Este trabalho descreve o desenvolvimento de um aplicativo para sistema Android a ser utilizado como plataforma de aquisição de dados em VANTs. É feita uma descrição detalhada do *software* desenvolvido, seguida pela análise de dados obtidos em testes. Avalia-se também o desempenho do sistema, comparando-o a outros sistemas de aquisição em termos de precisão e velocidade. Por fim é mostrado resultados da equipe Céu Azul Aeronaves na utilização deste sistema como auxílio no projeto de suas aeronaves.

## INTRODUÇÃO

Os *smartphones* representam uma das grandes evoluções da comunicação nas últimas décadas, extrapolando a aplicação de comunicação telefônica, para um dispositivo multitarefas, de organização pessoal, interação social, trabalho, educação e lazer. A possibilidade de ganhos de mercado levou a investimentos expressivos visando aumento de velocidade de processamento e memória e diminuição de custos, peso e dimensões [1]. Atualmente, o Android tornou-se o sistema operacional mais utilizado no mundo [2], sendo uma plataforma de fácil desenvolvimento e publicação de aplicativos por estudantes e programadores amadores.

Em especial, ao longo dos anos diversos sensores foram incorporados aos smartphones, entre os quais, acelerômetros, giroscópios, barômetros, magnetômetros e GPS [3]. Originalmente, esses sensores possuíam aplicação em funções básicas, como a detecção da rotação da tela e localização do usuário. Com o tempo, eles passaram a ser usados em outras aplicações gerando uma nova área de estudo e desenvolvimento conhecida por *Phone Sensing* [4]. Muitas áreas de aplicação na medicina, direito e engenharia, já se utilizam dessa tecnologia. Destacam-se projetos relacionados

ao monitoramento de saúde de pacientes [5], uso de dados dos sensores como evidência digital em aplicações forenses [6] e até mesmo envio de um *smartphone* para o espaço como *payload* em um satélite de pequeno porte [7].

Nesse contexto, este trabalho descreve a utilização de *smartphones* como uma plataforma de aquisição de dados em um veículo aéreo não tripulado (VANT), visando análise do comportamento de voo e validação de dados de projeto. Isso é realizado por meio do desenvolvimento de um aplicativo *open-source* para sistema operacional Android. O desempenho de diferentes *smartphones* e plataformas dedicadas para aquisição de dados em aeronaves são comparados entre si e com as informações dos fabricantes em [3] e [8], enfocando dimensões, facilidade de uso, preço, qualidade da informação e taxa de aquisição, expondo suas vantagens e limitações. O trabalho aborda apenas a aquisição dos dados, não sendo analisada a transmissão via rádio, opção também presente no aplicativo.

O artigo é dividido nas seguintes seções: *Fundamentos*, que expõe a arquitetura de *software* para obtenção de dados em dispositivos Android; *Metodologia*, que explicita ferramentas e técnicas utilizadas para o desenvolvimento; *Funcionamento*, que detalha o fluxo de processo do aplicativo; *Comparação entre smartphones*, que realiza comparações entre 3 diferentes dispositivos Android; *Comparação entre plataformas dedicadas*, que compara *smartphones* com duas outras plataformas dedicadas para aquisição de dados, e *Resultados*, que mostra dados obtidos com o aplicativo.

## FUNDAMENTOS

O sistema operacional Android possui uma estrutura de *software* padrão que permite instalar diferentes sensores em diferentes modelos de *smartphones* e acessá-los pelo *framework* do sistema [9]. A arquitetura de *software* para acesso aos sensores possui quatro níveis. O *kernel* é a camada mais baixa, responsável pelo acesso direto ao driver dos sensores através das interfaces de comunicação (*IPC*, em muitos casos). O *Hardware Abstraction Layer* (HAL)

funciona como máscara para o *hardware* [10]. O *framework* calcula parâmetros dos sensores, como taxa de amostragem máxima. Por fim, o *Software Development Kit* (SDK) dispõe de bibliotecas para o desenvolvimento de aplicativos [11] [9].

Os sensores são classificados de duas formas. A primeira, pela sua função: *Motion Sensors* medem movimento, como aceleração e rotação; *Environmental Sensors*, determinam parâmetros de ambiente, como temperatura, pressão e iluminação; *Position Sensors* calculam a posição física do dispositivo, como o sensor de orientação e o magnetômetro [12]. A segunda classificação é quanto a sua forma de geração da informação: um *Hardware Sensor* existe fisicamente e realiza medições diretas, como é o caso do acelerômetro; e um *Virtual Sensor* é criado virtualmente pela combinação dos dados de outros sensores por algoritmos de *sensor fusion*. Um exemplo dessa última categoria é o sensor de orientação, que utiliza o acelerômetro, giroscópio e magnetômetro para encontrar a orientação do dispositivo [9] [12].

Como a disponibilidade dos sensores é escolha do fabricante, cada modelo possui diferentes componentes. Entre os sensores mais comuns encontrados nos *smartphones* Android, destacam-se acelerômetro, termômetro, sensor de gravidade, giroscópio, iluminação, aceleração linear, campo magnético, orientação, pressão, proximidade, umidade, vetor de rotação, contador de passos e outros [12]. Cada um desses apresenta diferentes funcionalidades que podem ser usadas no sensoriamento de VANTs. Por exemplo, enquanto o acelerômetro mede a aceleração total nos 3 eixos do dispositivo (Figura 1), o sensor de aceleração linear descontará a componente da gravidade, mostrando apenas a ação causada por outras forças [12].

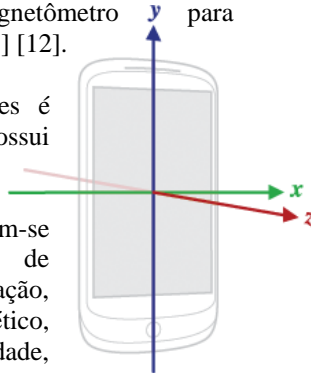


Figura 1 – Eixos do smartphone (Fonte [12])

Com exceção do GPS, os sensores no Android funcionam como uma *stream* de dados controlados por serviços externos. Em sua API, para cada amostra adquirida, dispara-se um sinal informando a existência de um novo dado, tratado pela interface *SensorEventListener* [10]. Nesse caso, a função *onSensorChanged()* é chamada, providenciando o objeto *SensorEvent*, que contém informações relevantes à modificação do valor [12]. Esta função deve possuir uma implementação leve tendo em vista que é chamada cada vez que um sensor adquire dado. Além disso, a API permite controlar a taxa de amostragem de cada sensor, dispondo de um *trade-off* entre velocidade e qualidade do dado ([8], [3]). Vale destacar que os sensores são calibrados pelo HAL, fornecendo dados já corrigidos na chamada das funções [9].

## METODOLOGIA

O aplicativo foi criado na interface de desenvolvimento *Android Studio 1.0.1* e compilado com a versão de SDK 21 (Android 5.0 - *Lollipop*), sendo mínimo SDK 14 (Android 4.0 - *Ice Cream Sandwich*), e está disponível na *Google Play* com o nome de Telemetria<sup>1</sup>.

Para o desenvolvimento do aplicativo utilizou-se uma adaptação para Android da arquitetura MVC (*Model-View-Controller*) como padrão de classes [13]. Como este aplicativo não requer uma interface gráfica complexa, as classes do tipo *controller* também são responsáveis pela exibição na tela (*view*). Um diagrama mostrando as principais classes pode ser visto na Figura 2. O código é inteiramente *open-source* e encontra-se no repositório do GitHub<sup>2</sup>.

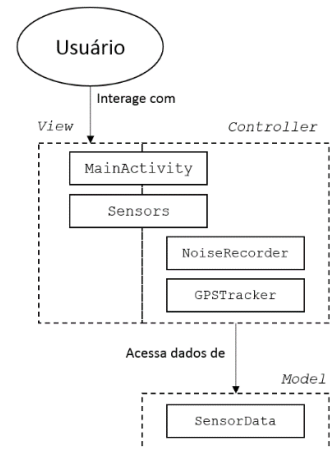


Figura 2- Principais classes do aplicativo e sua arquitetura



Figura 3- Tela inicial

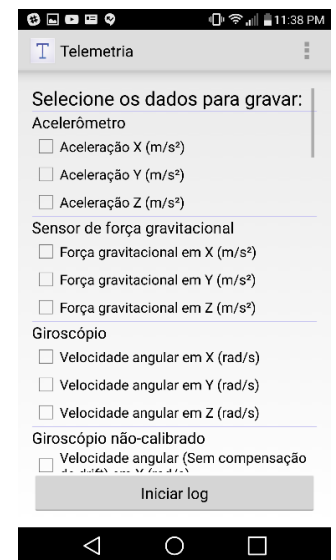


Figura 4- Tela de seleção de dados.

## FUNCIONAMENTO

O programa inicia na tela da Figura 3. Ao pressionar o botão “Iniciar” o código carrega a tela de sensores da Figura 4 e identifica os sensores presentes no *smartphone* com função *getSensorList()* [12]. Caso o GPS esteja habilitado, o código cria uma *thread* de calibração que funciona como um segundo

1 Download em <<https://play.google.com/store/apps/details?id=whitesun.telemetry>>

2 Código disponível em <<https://github.com/CeuAzul/Telemetry>>

processo, aguardando o sinal do GPS ser triangulado (sem bloquear o processo principal) e utiliza a primeira amostra latitude-longitude como referência para o ponto (0,0) no plano cartesiano. A partir deste ponto, todas as outras amostras de localização durante o voo serão convertidas para uma distância em metros ao ponto de referência, com base em uma aproximação da superfície terrestre por uma elipsoide WGS84 [12], visando facilitar as análises dos dados.

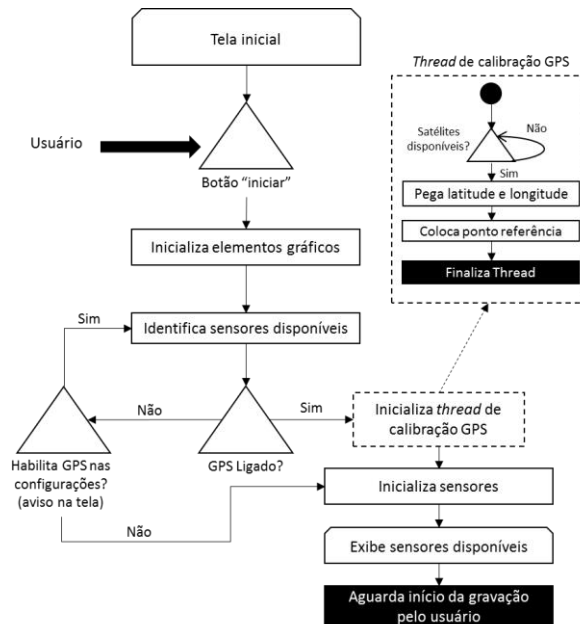


Figura 5- Diagrama de inicialização do sistema

Em seguida inicializa-se os sensores para que tenham sua aquisição de dados ouvidas pela função *onSensorChanged()*, e define-se a taxa de amostragem. Em consequência dos testes por [3] e [8], optou-se por realizar a aquisição na taxa mais rápida (*fastest*), aceitando-se, por exemplo, uma troca de 0,2 m/s<sup>2</sup> de acuracidade de aceleração por quatro vezes mais dados. Isso porque durante o voo, o próprio ruído atrapalha uma exatidão nesta proporção, sendo que a maior taxa de aquisição pode ajudar a filtrar o sinal em um pós-processamento.

Depois, o código carrega caixas de seleção de dados, juntamente com um campo para taxa de gravação. Por fim, aguarda-se o início da gravação pelo usuário. O fluxograma de inicialização pode ser visto na Figura 5.

O aplicativo utiliza a o formato CSV (*comma-separated values*) como estrutura de escrita. Este formato é amplamente utilizado para gravação de dados, tendo em vista sua simplicidade e independência [14]. O arquivo é escrito utilizando um cabeçalho com nome e unidade de medida em

cada coluna. Ao receber o comando do usuário, inicializa-se as *threads* de gravação do arquivo e de aquisição do GPS. Ambas só serão finalizadas quando o usuário decidir parar a aquisição pressionando o botão. Este processo é mostrado pela Figura 6.

É durante o processo de gravação que a função *onSensorChanged()* é tratada, identificando o dado que foi adquirido e armazenando-o em um objeto da classe *SensorData*. Como o acesso dos dados do GPS acontece de forma diferenciada, uma *thread* é criada especificamente para isso. Esse processo periodicamente verifica alteração no valor de localização, e, em caso positivo, calcula a posição em relação ao ponto de referência.

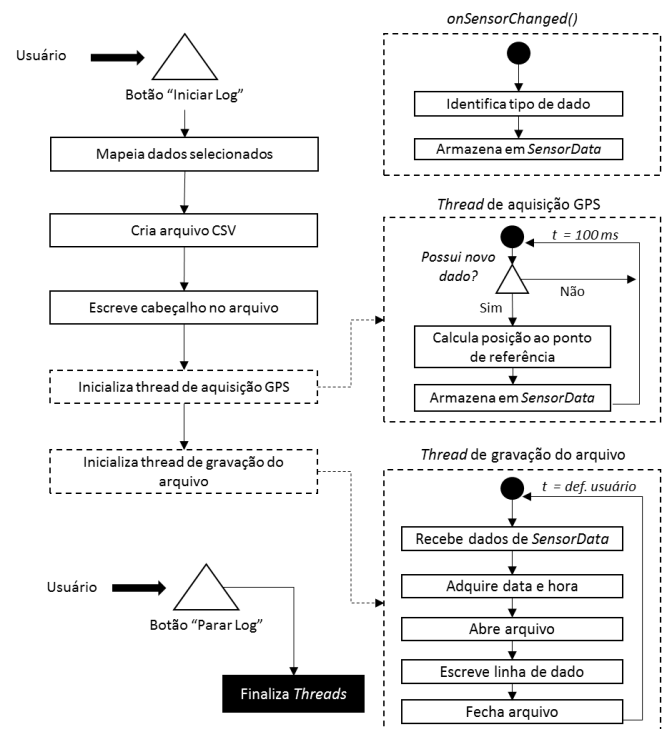


Figura 6 - Diagrama de fluxo de gravação

A escrita do arquivo também acontece em uma *thread* separada. Os dados que foram armazenados nos objetos *SensorData* são lidos juntamente com a referência de tempo. Neste processo, abre-se o arquivo CSV (*FileWriter.open()*), onde é gravada a linha de dados. Immediatamente em seguida o arquivo é fechado (*FileWriter.close()*). Destaca-se a estratégia de abrir e fechar o arquivo em cada linha escrita é um processo que requer mais tempo de processamento, mas garante que não haja perda de dados caso o celular desligue ou danifique - desconectando a bateria em uma queda de avião, por exemplo - já que os dados passam de um *buffer*, para um arquivo no cartão SD [15].

O processo de gravação é executado em uma taxa definida pelo usuário, que pode ser ajustada de acordo com a necessidade do teste de voo e das características de processamento do *smartphone*. A taxa de gravação pré-definida também facilita a filtragem de dados, já que mantém constante intervalo de tempo entre cada amostra facilitando o uso de uma FFT (*Fast Fourier Transform*), por exemplo.

## COMPARAÇÃO ENTRE SMARTPHONES:

Modelo	Galaxy S	Motorola G1	LG-G3
Processador	Samsung Exynos 3 (ARM 1.0GHz Quad-Core)	Cortex-A7 (1.2 GHz quad-core)	Krait 400 (2.5 GHz quad-core)
Sistema operacional	Android 2.3 "Gingerbread"	Android 5.1 "Lollipop"	Android 6.0 "Marshmallow"
Ano	2010	2013	2014
Memória	512MB RAM	1 GB RAM	3 GB RAM
Sensores detectados	BMA023 Acceleration	LI53DH Accelerometer	LGE Accelerometer
	MS-3C Magnetic Sensor	UK8963 Magnetic field	LGE Magnetic Field sensor
	MS-3C Orientation Sensor	AK8963 Orientation sensor	LGE Gyroscope
	GP2A Light Sensor	CT406 Proximity sensor	LGE Proximity sensor
	GP2A Proximity Sensor	CT406 Light sensor	LGE Light sensor
	Gravity Sensor	Display Rotation sensor	LGE Pressure sensor
	Linear Acceleration Sensor	GPS	LGE Gravity Sensor
	Rotation Vector Sensor	Microphone	LGE Linear Acceleration
	GPS		LGE Rotation Vector Sensor
	Microphone		LGE Step Detector Sensor
			LGE Motion Detector Sensor
			LGE Orientation Sensor

Tabela 1 - Características dos smartphones testados.

Visando analisar o desempenho do aplicativo utilizou-se três diferentes *smartphones* como plataformas teste. Buscou-se utilizar dispositivos de gerações diferentes, distinguindo-se em velocidade de processamento, memória e versão de sistema operacional, tal como descrito na Tabela 1. As comparações são feitas com base em 3 critérios: precisão, faixa de medição e taxa de aquisição. Para verificar a precisão de cada sensor utiliza-se a função *getResolution()* da classe *SensorManager* e o *range* (faixa de medição) é obtida por *getMaximumRange()* [12]. A Tabela 2 mostra o compilado dessas informações para alguns dos sensores.

Verifica-se pelas Tabelas 1 e 2 que alguns dispositivos possuem limitações importantes para aquisição de dados em aeronaves. Por exemplo, o *smartphone* Galaxy S é o que possui menor resolução em seus sensores, chegando a erros de 1,406° ao medir ângulos de *yaw*, *pitch* e *roll*. Outra limitação é quanto a faixa máxima de medição. O acelerômetro do Galaxy S satura seus valores em aproximadamente  $\pm 2g$ , enquanto Motorola G1 e o LG-G3 alcançam valores próximos de  $\pm 4g$ . Deve-se atentar a estes valores ao realizar a validação das forças presentes durante o voo, tendo em vista que estas podem saturar e gerar erros no projeto.

Resolução máxima			
	Galaxy S	Motorola G1	LG-G3
Acelerômetro	0,0383 m/s <sup>2</sup>	0,00981 m/s <sup>2</sup>	0,00119 m/s <sup>2</sup>
Giroscópio	-	-	0,00107 °/s
Magnetômetro	0,585 $\mu$ T	0,0625 $\mu$ T	0,149 $\mu$ T
Orientação	1,406 °	0,0156 °	0,100 °
Barômetro	-	-	0,0131 hPa
Faixa máxima de medição			
	Galaxy S	Motorola G1	LG-G3
Acelerômetro	19,613 m/s <sup>2</sup>	39,240 m/s <sup>2</sup>	39,226 m/s <sup>2</sup>
Giroscópio	-	-	34,907 °/s
Magnetômetro	300,000 $\mu$ T	2000,000 $\mu$ T	2400,000 $\mu$ T
Orientação	360 °	360 °	360 °
Barômetro	-	-	1100 hPa

Tabela 2 - Propriedades dos sensores

Para encontrar a taxa máxima de gravação, coloca-se os dispositivos para adquirirem dados de aceleração, orientação, campo magnético e localização (GPS) na maior velocidade possível (*delay* = 0 entre cada linha de dado do arquivo CSV). Utilizou-se o aplicativo para gravar 10.000 linhas de dados para cada dispositivo e o tempo ( $\Delta t$ ) de gravação entre elas foi calculado (conforme visto na Figura 7). O resultado dos testes pode ser visto na Tabela 3.

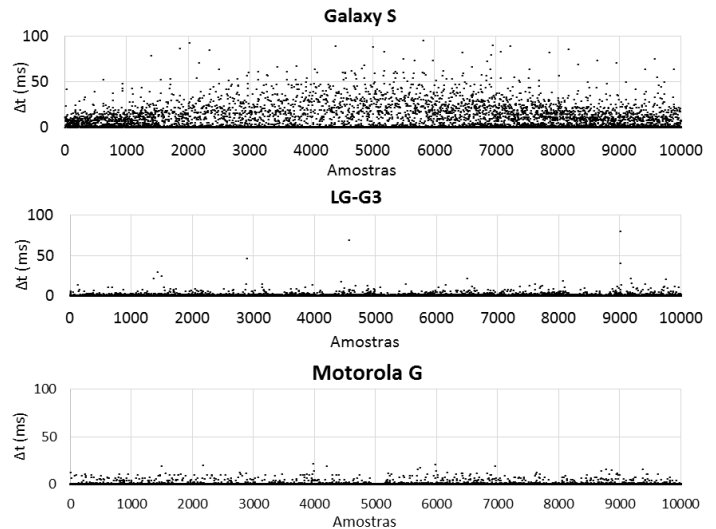


Figura 7 - Tempo de escrita de uma linha de dado na velocidade máxima.

	Galaxy S	Motorola G	LG-G3
Delay médio (ms)	6,625	0,762	1,388
Desvio padrão (ms)	11,210	1,455	11,648
Delay máximo (ms)	101,000	22,000	687,000
Delay mínimo (ms)	< 1	< 1	< 1
Taxa de gravação média (Hz)	150,943	1312,853	720,565

Tabela 3 - Taxa máxima de gravação de uma linha de dado.



A taxa de aquisição de cada sensor pode ser obtida por meio de um teste similar, calculando a diferença de tempo entre duas aquisições e gerando uma nuvem de pontos. Este teste foi feito para os sensores mencionados acima durante 1 minuto de gravação e os resultados mostrados na Figura 8. Nela, compara-se os valores médios obtidos no teste (experimental) com os dados fornecidos pelo fabricante pela função *getMinDelay()* [12].

Durante testes, a taxa de gravação recomendada deve ser maior que a taxa de amostragem dos sensores, visando não perder informações [16], mas deve ser menor que a taxa máxima de gravação, para que não haja informações repetidas, consumo excessivo de bateria e instabilidade no processamento [17]. Assim a Tabela 4 mostra uma taxa de gravação recomendada para cada um dos dispositivos.

	Galaxy S	Motorola G	LG G3
Taxa de gravação recomendada (Hz)	50,000	100,000	100,000

Tabela 4 - Taxa de gravação recomendada.

## COMPARAÇÃO COM PLATAFORMAS DEDICADAS

Modelo	T2015	Pixhawk PX4
Processador	ATmega2560 16MHz	ARM Cortex M4F 168 MHz
Memória	8 KB SRAM, 4 KB EEPROM, 256 KB Flash	256 KB RAM, 2 MB Flash
Software	NiRTOS	PX4 software
Sensores	MPU 6050 Gyroscope	L3GD20H Gyroscope
	MPU 6050 Accelerometer	LSM303D Accelerometer
	HMC5883L Magnetometer	LSM303D Magnetometer
	HC-SR04 Ultrasound sensor	MPU 6000 Accelerometer
	Infrared Tachometer	MPU 6000 Gyroscope
	MS5611 Barometer	MS5611 Barometer
	MS5611 Thermometer	MS5611 Thermometer
	MP3V5004GP Pressure sensor (Pitot probe)	4525DO Pressure sensor (Pitot probe)
	U-blox Neo 6M GPS	U-blox Neo-M8N GPS

Tabela 5 - Características técnicas das plataformas de aquisição de dados utilizadas para comparação.

Compara-se o desempenho do aplicativo frente a duas diferentes categorias de plataformas dedicadas de aquisição de dados. A primeira delas, T2015 foi desenvolvida pela equipe Céu Azul Aeronaves utilizando *Arduíno Mega*, ao participar da competição SAE AeroDesign na classe *Advanced*, construída no estilo *do-it-yourself* [18]. A segunda, *Pixhawk*, é uma placa de aquisição de dados e controladora, muito

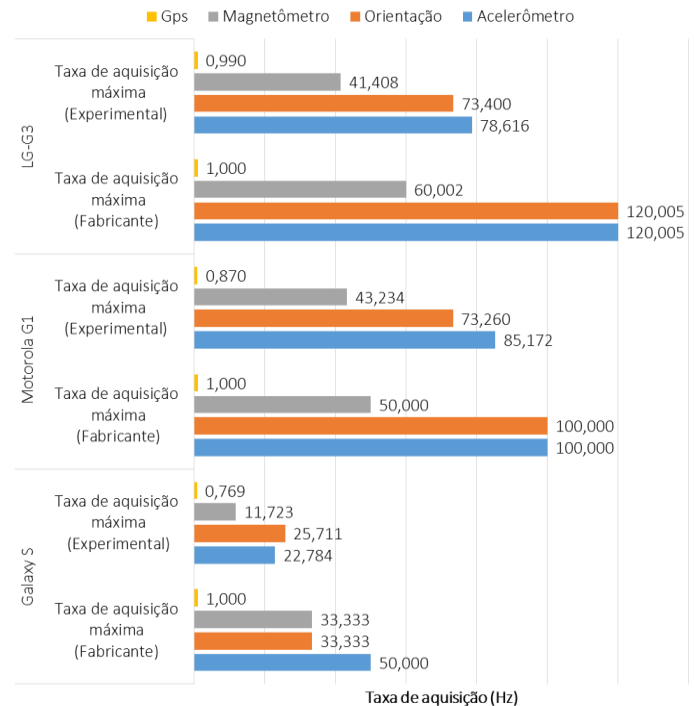


Figura 8 - Taxa de aquisição dos sensores

utilizada comercialmente em projeto de drones autônomos de mapeamento, ambas vistas na Tabela 5. Os critérios de comparação foram os mesmos utilizados em [19].

A Tabela 6 mostra a comparação das plataformas com o um *smartphone* LG-G3. Tanto a massa quanto o volume são estimativas da soma das placas principais com baterias e sensores. Parâmetros de precisão e *range* foram retirados de [20], [19] e do *datasheet* dos sensores. Para T2015 e para o LG-G3, a taxa de gravação foi obtida experimentalmente, e para o *Pixhawk* utilizou-se o valor mostrado em [19]. Tempo de instalação da T2015 e do LG-G3 foi obtido durante testes experimentais, e para o *Pixhawk* com base em uma pesquisa com utilizadores da plataforma.

Verifica-se pela tabela que os *smartphones* possuem pouco volume e dimensões reduzidas, o que facilita a instalação em aeronaves pequenas devido ao seu fácil encaixe no compartimento de carga. Tendo em vista seu processador de 2,5 GHz, a taxa de aquisição do *smartphone* é o dobro da *Pixhawk* e cerca de 5 vezes maior que a T2015.

Devido à janela de tempo limitada de voos testes em aeroclubes, facilidade de uso e tempo de instalação tornam-se aspectos fundamentais. A utilização do aplicativo em um *smartphone* é extremamente simples, bastando selecionar os dados que se deseja gravar e fixá-lo na aeronave. Em geral, *smartphones* possuem bastante espaço em sua memória interna para gravação do arquivo e não necessitam de cabos, já

que o arquivo pode ser mandado para o e-mail logo após o voo, ou mesmo compartilhado pelo *WhatsApp*. Ademais, *smartphones* eliminam a necessidade de qualquer conhecimento de eletrônica para adquirir dados dos aviões, diferente da T2015 e da *Pixhawk*, que requerem manutenção de seus conectores e soldas. Em caso de problemas, *smartphones* são muito mais comuns que as outras plataformas, tendo em vista que o aplicativo possui (segundo *Google Play Console*) compatibilidade com mais de 7930 diferentes modelos de *smartphones*.

	T2015	Pixhawk	LG-G3
<b>Massa</b>	0,485 kg	0,164 kg	0,149 kg
<b>Dimensões</b>	21,50 cm	5,00 cm *	14,6 cm
	11,30 cm	8,15 cm	7,4 cm
	5,5 cm	1,15 cm	0,89 cm
<b>Volume total</b>	1336,22 cm <sup>3</sup>	82,14 cm <sup>3</sup>	86,43 cm <sup>3</sup>
<b>Tempo de instalação</b>	10 min	7 min	3 min
<b>Taxa de gravação</b>	20 Hz	50 Hz	100 Hz
<b>Acelerômetro (Range)</b>	±16 g	±16 g	±4 g
<b>Magnetômetro (Range)</b>	±800 µT	±1200 µT	±2400 µT
<b>Giroscópio (Range)</b>	±2000 °/s	±2000 °/s	±34,907 °/s
<b>Barômetro (Precisão)</b>	0,012 hPa	0,0115 hPa	0,0131 hPa
<b>GPS (Taxa)</b>	1,25 hz	5 hz	1 hz
<b>Velocidade do ar (Pitot) (Range)</b>	80 m/s	100 m/s	-
<b>Proximidade (Range)</b>	20 cm	-	5 cm

Tabela 6 - Comparação entre diferentes plataformas de aquisição com o smartphone LG-G3.

Os valores de *range* tanto do acelerômetro quanto do giroscópio são menores que das outras duas plataformas. Apesar disso, em condições normais de voo ambos são limitantes razoáveis para os valores nas aeronaves - não descartando cuidado na hora de analisá-los. Uma comparação dos valores de aceleração da T2015 e do Galaxy S pode ser visto na Figura 9. Além disso, tanto o barômetro quanto o magnetômetro possuem características similares nas plataformas, como mostrado na Figura 10.

O *smartphone* perde em adaptabilidade de *hardware*. Tanto a T2015 e a *Pixhawk* possuem portas adicionais configuráveis, como *SPI*, *I<sup>2</sup>C* ou *UART* ou entradas analógicas [20], permitindo adicionar outros sensores, como o tacômetro óptico presente na T2015. Isso também mostra a maior desvantagem dos *smartphones* para aquisição de dados de aeronaves: não possuir transdutor de pressão para obter a velocidade do ar por meio de um tubo de *pitot*. A aproximação deste parâmetro pela velocidade do GPS pode gerar erros devido à baixa taxa de aquisição, conforme a Figura 11. Em termos de custos, (Tabela 7) *smartphones* com um ou dois

anos de uso são uma ótima opção para serem usados como uma plataforma dedicada para aquisição de dados. Vale destacar que a utilização desses dispositivos é uma ótima aplicação para os *smartphones* no final de sua vida útil, proporcionando uma opção de reciclagem. No caso da equipe

Céu Azul Aeronaves, todos os *smartphones* utilizados para aquisição são doados pelos próprios membros, quando compram um novo modelo, ou quando ocorre problemas não relacionados com o sensoramento (uma rachadura na tela por exemplo).

	Faixa de preço	
<b>T2015</b>	R\$ 150,00	R\$ 250,00
<b>Pixhawk PX4</b>	R\$ 650,00	R\$ 900,00
	Usado	Novo
<b>Galaxy S</b>	R\$ 80,00	R\$ 350,00
<b>Motorola G</b>	R\$ 95,00	R\$ 460,00
<b>LG-G3</b>	R\$ 150,00	R\$ 730,00

Tabela 7 - Comparação de preço entre as plataformas

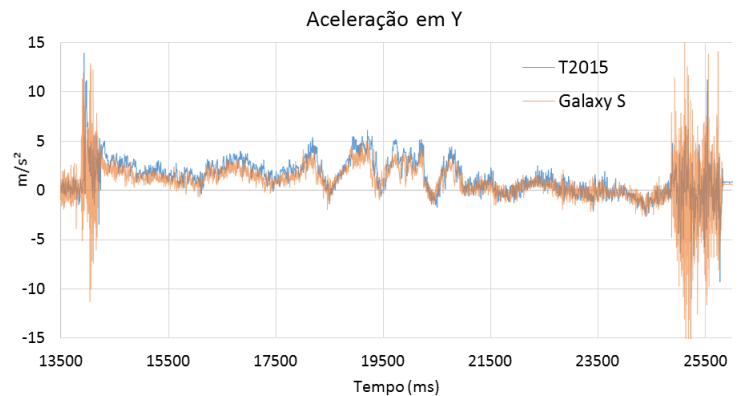


Figura 9 - Comparação dos valores de aceleração obtidos pelo aplicativo (no Galaxy S) com o sistema de aquisição T2015 durante um teste de recuperação de estol com um avião da categoria regular.

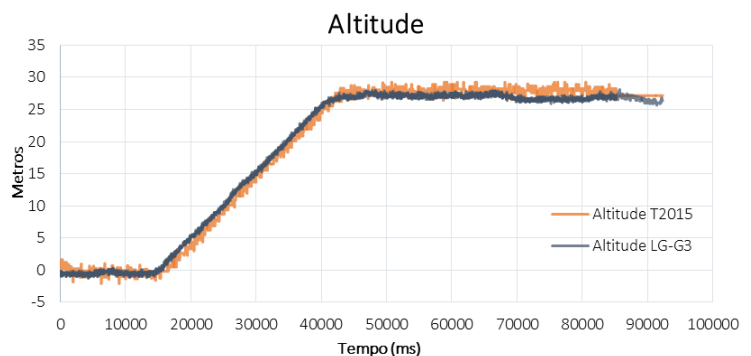


Figura 10 - Comparação entre o barômetro do LG-G3 e da T2015 funcionando como altímetro durante uma subida de elevador em um prédio de 9 andares.

## RESULTADOS

Os dados dos sensores dos *smartphones* podem ser utilizados de inúmeras formas em projetos de VANTs. A Figura 12

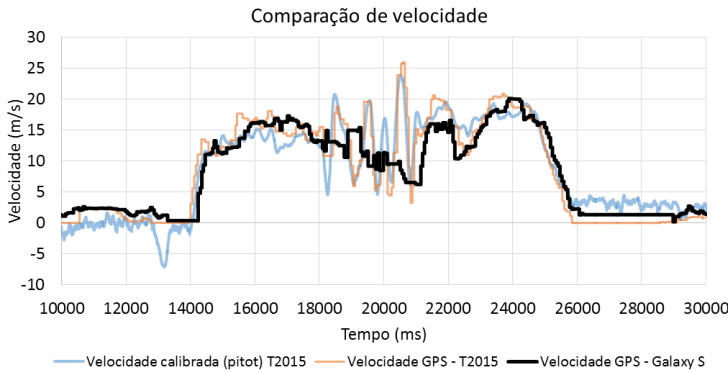


Figura 11 - Diferentes tipos de aquisições para a velocidade de um avião da categoria regular.

mostra a orientação e trajetória de uma aeronave durante voo, servindo para análises de estabilidade e desempenho. A Figura 13 indica o momento da queda de uma aeronave da classe regular. A posição da queda assim como a velocidade e aceleração são mapeados pelo dispositivo auxiliando na identificação e no diagnóstico de falhas. Medições como as forças presentes no voo podem ser adquiridos pelos sensores de aceleração (Figura 9) e podem validar dados de cargas, como o diagrama V-N. A precisão do barômetro permite medir variações sutis de altitude (Figura 10), possibilitando a obtenção experimental da razão de subida e descida. Outras medições, como identificação do momento de decolagem pode ser feita pelo sensor de proximidade assim como as velocidades de cada etapa de voo são validadas pelo GPS.

## CONCLUSÕES ESPERADAS

Verificou-se que os dispositivos mais recentes (Motorola G e LG-G3) possuem resolução suficientemente boas para a gravação de dados em voo gerando um erro de amplitude pequeno comparado ao ruído durante voos normais. Ademais, tanto precisão quanto range possuem valores comparáveis às plataformas dedicadas. Devido ao alto poder de processamento, *smartphones* podem gravar dados em taxas até duas vezes maiores que plataformas de aquisição comerciais. Das principais desvantagens verifica-se a baixa taxa de aquisição do GPS e dificuldade na adaptação da plataforma para receber novos sensores, destacando-se a não existência de transdutores de pressão para medição da velocidade de escoamento do ar. Das vantagens, destaca-se a grande disponibilidade desses dispositivos no meio social, seu volume e peso reduzido, tempo de instalação, e principalmente, sua enorme facilidade de uso.

Conclui-se que os *smartphones* são uma ferramenta com grande potencial para sensoriamento em aeronaves. O *software* apresentado nesse trabalho se encontra disponível

como um aplicativo *open-source*. Espera-se no futuro que esse aplicativo possa ser utilizado e melhorado por mais equipes tornando-se um elo entre competição e cooperação entre as equipes de AeroDesign.

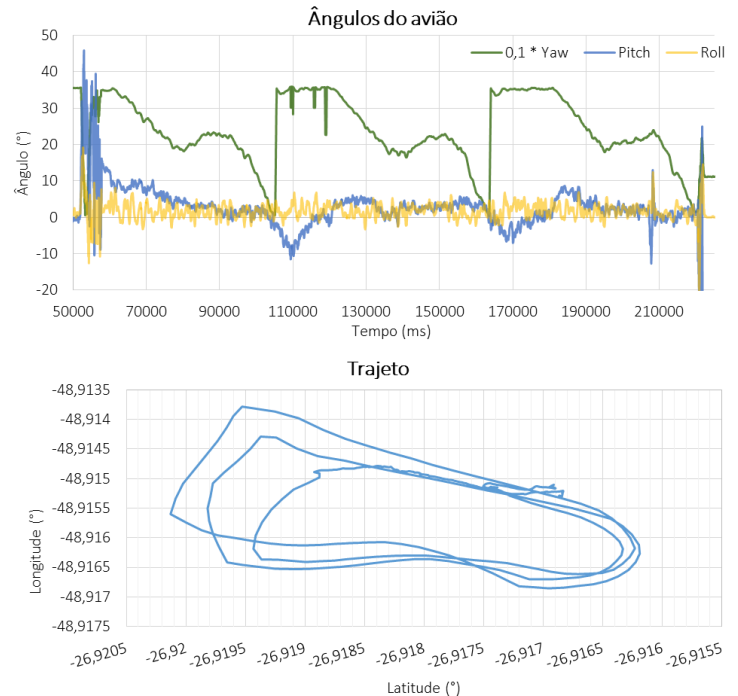


Figura 12 - Dados de 3 voltas de um avião da categoria Regular usando o aplicativo em um Samsung Galaxy S.

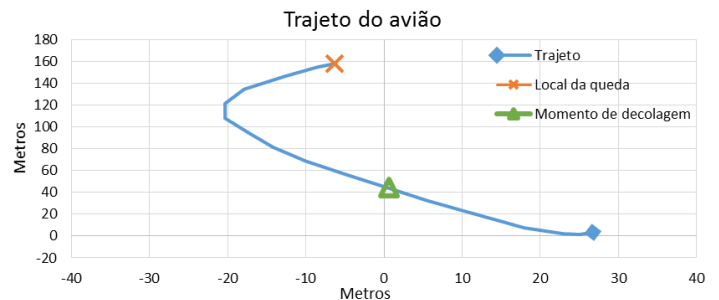
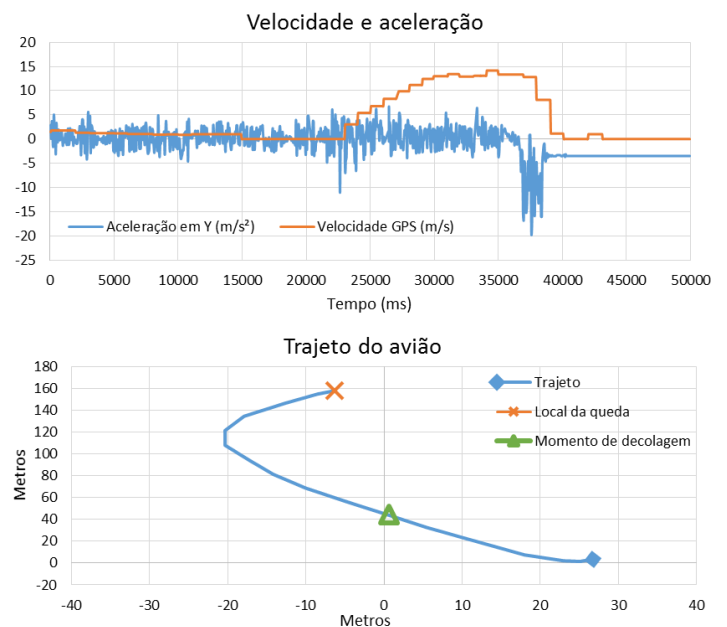


Figura 13 - Dados da queda de um avião da classe Regular medidos por um Galaxy S.

### REFERÊNCIAS

1. YANG, et al. **Crowdsourcing to Smartphones: Incentive Mechanism**. 18th annual international conference on Mobile computing and networking. Istanbul, Turkey: [s.n.]. 2012. p. 12.
2. G1. Tecnologia e Games. **Android passa Windows e se torna o sistema operacional mais usado do mundo**, 2017. Disponível em: <<https://g1.globo.com/tecnologia/noticia/android-passa-windows-e-se-torna-o-sistema-operacional-mais-usado-do-mundo.ghtml>>. Acesso em: 15 fev. 2018.
3. ZHIZHONG, M. **Experimental Evaluation of Mobile Phone Sensors**. ISSC 2013.: 2013. p. 9.
4. LANE, N. et al. **A Survey of Mobile Phone Sensing**. IEEE Communications Magazine: IEEE. 2010. p. 11.
5. NANHORE, S. D.; BARTERE, M. M. **Mobile Phone Sensing System for Health Monitoring**. International Journal of Science and Research: 2013. p. 4.
6. MYLONAS, A. et al. Computers & Security. **Smartphone Sensor Data as Digital Evidence**, v. 38, p. 51-75, Outubro 2013.
7. C. P. BRIDGES, Y. B. et al. **Smartphone Qualification & Linux-based Tools for CubeSat Computing Payloads**. IEEE., p. 10. 2013.
8. SHALA, U, et al. **Indoor Positioning using Sensor-fusion in Android Devices**. School of Health and Society. Kristianstad: set. 2011.
9. QUALCOMM TECHNOLOGIES, INC. **Sensors Porting Guide**. p. 18. 2015.
10. YIN YAN et al. **Real-Time Sensing on Android**. The 12th International Workshop on Java Technologies for Real-time and Embedded Systems. Buffalo: Outubro 2014.
11. ANDROID OS. Source Android. **Sensor stack**, 2017. Disponível em: <<https://source.android.com/devices/sensors/sensor-stack>>. Acesso em: 28 jan. 2018.
12. ANDROID. **Android Developer**, 2018. Disponível em: <<https://developer.android.com>>. Acesso em: 28 jan. 2018.
13. SOKOLOVA, K. **Android Passive MVC a Novel Architecture Model**. The Fifth International Conferences on Pervasive Patterns and Applications: 2013. p. 6.
14. MITLOHNER, J. et al. **Characteristics of Open Data CSV Files**. 9th International Conference on Semantic Systems. Vienna, Austria: IEEE. 2016. p. 8.
15. BUELL, D. **Data Structures using Java**. Burlington: Jones & Bartlett Learning, 2013.
16. BERSCH, S. D. et al. **Sensor Data Acquisition and Processing Parameters for Human**. ISSN 1424-8220. 2014. p. 32.
17. MISRA, A.; LIM, L. **Optimizing Sensor Data Acquisition for Mobile Data Management (MDM)**. Lulea, Sweden: IEEE. 2011. p. 10.
18. SOUSA, R. R. D.; MARGI, C. B. **Sensoreamento sem fio em aeromodelos radiocontrolados**. Escola Politecnica da USP. Sao Paulo / SP, p. 43. 2013.
19. D. DANTSKEER, R. M. et al. **High-Frequency Sensor Data Acquisition System (SDAC) for Flight Control and Aerodynamic Data Collection**. 32nd AIAA Applied Aerodynamics Conference. Atlanta, GA: AIAA Aviation. 2014. p. 17.
20. PIXHAWK. **Pixhawk Autopilot**. Disponível em: <<https://pixhawk.org>>. Acesso em: 06 fev. 2018.

### CONTATO

**Leonardo Mariga** – Universidade Federal de Santa Catarina  
leomariga@gmail.com

**Rafael Araújo Lehmkuhl** – Universidade Federal de Santa Catarina  
rafael.lehmkuhl93@gmail.com

**Amir Antonio Martins Oliveira Jr** – Universidade Federal de Santa Catarina  
amir.oliveira@gmail.com