

IMU-Centric Multi-Sensor Fusion via Factor Graph Optimization: A Review of Robust Navigation and Adaptive Weighted Fusion

Jiawen Su^{1*†}, Yixin Jing^{2†}, Maisha Lu^{3†}, Xingyu Pan^{4†}

¹*School of Precision Instrument and Opto-electronics Engineering, Tianjin University, Tianjin, China*

²*China National Electric Apparatus Research Institute Co.Ltd. Sun Yat-sen University, Guangzhou, China*

³*Mechanical Engineering Department, Dalian Maritime University, Dalian, China*

⁴*Mechanical and Electronic Engineering, Beijing Jiaotong University, Beijing, China*

**Corresponding Author. Email: sjwen136@163.com*

†These authors contributed equally to this work and should be considered as co-first authors

Abstract. Precise and dependable positioning is needed for a lot of different areas such as self-driving cars, robot, and autonomous systems especially in complicated or not good situations. To get rid of the limitations of monolithic sensor architecture, the utilization of multiple sources has emerged as an important method and FGO is getting more recognized as a good framework for estimating states because it has global coherence and flexible structure. In this paper, we do a full examination of the IMU-centered multimodal integration frameworks in the FGO setup involving IMU-GNSS, IMU-vision, IMU-LiDAR-vision and IMU-4D radar. Examining all the pros and cons of each kind of sensing, especially looking into the way that working together makes it more reliable and precise. Furthermore there is also an extra classification of adaptive integration protocols referred to as model-based vs data-based it shows about what the protocol does when a failure or variance in the sensor occurs. According to the survey it is a major way of strengthening systems Adaptive Weighted Fusion (AWF) but what currently exists remains mostly sensor based and disjointed. At last this paper talks about remaining troubles and possible examination route, it stresses the need of consistent adaptive structure can deal different sensors in practical surroundings.

Keywords: Multi-Sensor Fusion, Factor Graph Optimization

1. Introduction

It is still important for the application such as autonomous driving, mobile robots and unmanned aerial vehicles to have reliable and accurate navigation. To resolve that we do something called multisensor integration in other words it is using multiple type of sensor data to make a more accurate and robust solution [1]. Among the present approaches FGO turns out to be an effective

method of state estimation. Using node on those graphs for states of systems and observables(Variables), the FGO shall also optimize again over joints at a joint level as it is global over multiple sensors [2]. Of course, in those cases, it is the IMU that becomes something of an important part due to its ability to provide high-rate motion data unconstrained. The IMU readings rely solely on internal sensing so its not affected by outside light or weather nor things in the way which is good for places that are challenging indoor environments [3]. But it also has major restrictions, like unbounded drift, temperature-induced biases, and noise accumulation, so it gets very wrong after a long time [4]. In order to solve this problem, it has generally some external sensor that gives a extra constraints so that we can rely more longer term. Recently there is more attention on the combination of 4D radar with IMU. Combining radar Doppler information along with the inertial motion signal leads us to more strong and robust way of estimating motions even in challenging navigation environments which can be very promising step toward reliable navigation [5]. Looking forward, it is certain that many kinds of the integration system has more flexible with a closer connection. Concerning design to come: We will need focus on dynamic sensor weights, cross-modal learning and the like; to cope with unstructured situations which keep changing.

2. Multi-sensor fusion

2.1. Robust IMU-GNSS fusion via factor graph optimization

As far as helping out with some other sensors go, GNSS is easily the most commonly used sensor in combination with an IMU for outdoor navigation. This gives a kind of absolute position reference that can bound the long-term integration drift of purely inertial measurement. Urban canyons have lots of multipath propagation and NLOS reception which messes up the measurement of the pseudorange by like 10s of meters [6]. Traditional filter-based methods set a single number for the whole GNSS position fix – a single bad satellite can make things really bad but never be noticed; factor graph optimization keeps those raw per-satellite pseudorange observations separate & independent, which lets you model noise in lots of small bits and selectively give lower importance to those that don't look right [7].

Table 1. IMU-GNSS methods summary

Method	Ref.	Coupling	Core innovation	Calib.	Key result
SLAC-FGO	[7]	LC async	EM Gaussian mixture + joint calib	Yes	0.233 m RMSE
RFGO	[8]	TC	Barron adaptive loss	No	41% RMSE↓ vs FGO
Raw-GNSS FG	[9]	TC+CP	Double-diff carrier phase	No	Matches cam+LiDAR
ASW-FGO	[10]	LC+ODO	Adaptive window + ODO factor	Yes	75.8% faster

Fusing IMU and GNSS through FGO has become the most popular way to do outdoor positioning. The big difference from all others is in assigning the one and only weight it took when receiving a position fix. In the FGO we had just taken each pseudorange to their respective satellite, doing this on every inch but optimizing them [8]. This is a satellite specific thing to characterize the noise: The amount of noise given to each GNSS variable in the model is what really controls the precision more than the graph. The current methods address it with the two views taken in this section, data driven uncertainty estimation is covered by section III.

Robust noise modeling

The traditional FGO is based on the Gaussian pseudorange noise assumption which will be invalid if NLOS data gives rise to severe deviations. Ahmadi et al. [8] used combination of squared-error loss on all the GNSS variables alongside Generalized robust loss w/Shape (Which varies between L2 to Cauchy/Welsch depending on what is observed). And they tuned according to which one fits better to what can be seen The RMSE for the UrbanNav Hong Kong data was reduced by 41% RFGO and 50% from EKF. Bai et al. [7] took the probability, they used a Gaussian mixture for pseudorange discrepancy which they estimated on-the-go with EM. Residuals from NLOS reflection were given to the high variance group and then down weighted again without any threshold, this same algorithm also calculated the extrinsic calibration variables in one go along with SLAC (Joint Localisation & Calibration) giving an RMSE of 0.233 m Horizontal. In summary all these say that it's not about how we modify the structure of the graph but more on picking a good noise model.

Table 2. IMU-CAMERA methods summary

Method	Ref.	Coupling	Core innovation	Calib.	Key result
IMU Pre-integration FGO	[11]	TC	On-manifold IMU pre-integration in a sliding-window factor graph, jointly optimizing poses, velocities, and biases	No	high-rate IMU data into compact motion constraints
OKVIS	[12]	TC	Early keyframe-based visual-inertial FGO with nonlinear optimization over visual reprojection and IMU residuals	No	real-time VIO in UAV
VINS-Mono	[13]	TC	Monocular tightly coupled VIO with loop closure, relocalization, and online camera-IMU extrinsic estimation	Yes	Achieved real-time performance for UAV applications
ORB-SLAM3	[14]	TC	Unified visual / visual-inertial SLAM with multi-map support, loop closure, and strong relocalization capability	No	3.5 cm accuracy on EuRoC and 9 mm accuracy
Kimera	[15]	TC	Tightly coupled VIO extended with metric-semantic mesh reconstruction	No	Enables 3D accurate state estimation
Time-Offset / Rolling Shutter Optimization Framework	[16]	TC	Optimization-based modeling of varying camera-IMU time offset and rolling-shutter distortion	Yes	Improves robustness of smartphone visual-inertial fusion

Measurement extension and adaptive window management

Beuchert et al. [9] broadened what could be observed besides pseudorange by adding double-differenced carrier-phase terms. Distinguishing the 2 sat's and 2 epochs removes clocks issues & big atmos delay, resolve int ambig...no baseline: Pseudorange gives us the absolute geodetic basis, carrier phase adds sub-decimeter relative constraint. A driver took a 41-minute car ride through town it was equally precise as a camera + GNSS-IMU-LiDAR combo which nobody would be able to do just by making up some distance for themselves. Ji et al. [10] tried to break the compromise on fixed window, ASW-FGO's longer windows during GNSS outage are longer than when it was observed by Schur complement marginalization. Odometer velocity has an ongoing guess about both mount and scale. The results of the 193 second urban outage test shows that ASW-FGO was an exact 200-second fixed window however it did so in 75.8% less processing time.

Review of prominent IMU-GNSS Integration Techniques that have been reviewed in this paragraph can be found in Table I:

There's an element of the carrier phase with something missing, plus adaptive noise: resilient losses have, not looked at in carrier deviation – maintaining the integer ambiguity structure makes it hard to tell apart. It's fusing up adaptive noise along with online calibration of carrier phases for the infrastructure-free high precision.

2.2. IMU-CAMERA fusion via factor graph optimization

In the IMU–vision system, the FGO is the state at each and every keyframe:

$$x_k = \{ R_k, p_k, v_k, b_k^g, b_k^a \} \quad (1)$$

Put IMU pre-integration in the same window as visual reprojection errors and loop-closure constraints: The IMU pre-integration condenses the high frequency info of inertia into a short relative motion constraint, then fast optimization [11]:

The most important are the keyframe driven visual-inertial Structure OKVIS [12], the Monocular VIO Tightly Integrated With Loops OF VINS-Mono [13] The Unified Visual inertial System 3 from ORB-SLAM makes multiple map using orbs-lm 3 [14] And kimera which uses VI with semantic mesh reconstruction [15] FGO is different from Kalman in that FGO linearizes the residuals with respect to a sliding window rather than globally and therefore can much more easily concurrent optimize over poses and velocities, sensor offsets and extrinsics, loop closure and relocalization: But it's still need a lot of calculate force and memory.

To compare with filter, it can also be preferred if you are resource-constrained platforms.

4Seasons dataset can be seen as the best resource to test autonomous driving systems because of its variety in roads and atmospheres [17]. This is a good place to put things like testing how long they work relocalize ORB-SLAM3's multi-map module plus loop closures let you find your way after bad eyesight, but stereo + IMU is better for cities and highways than monocular + IMU. The good of IMU-vision integration is due to the merits made by the combination of cameras and IMUs: cameras give lots of structural information while IMUs keep things consistent when occluded or GNSS has problems. Constraints are big look changes, yearly textures design changes and sparse visual features under high-speed motion.

With respect to the UAVs, there is an EuRoC MAV dataset provides synchronized stereo imagery along with the IMU info for assessing quick motion as well as major changes in orientation [18].

OKVIS and VINS-Mono had real time ability on UAV. ORB-SLAM3 has a precision of 3.5 cm in stereo - inertial SLAM on EuRoC which shows that FGO can be very precise [14]. IMU-vision integration can reduce the fast movements and temporary vision blocking that causes problems when there is no GNSS signal, but it has trouble like having pose oscillation and feature tracking loss, especially for monocular systems.

For portable hardware, handheld devices and so on. TUM VI benchmark is just what we need to evaluate VIO's efficiency when there is fast movement and large local parallax [19]. With regard to OKVIS and ORB-SLAM3 the handheld trajectory was very good. On room sized datasets ORB-SLAM3 is around 9 mm: Kimera goes on to construct the metric – semantic Mesh using 3D rendering which is important for Wearable Spatial Awareness And indoor Digital Twin System [15]: However, there are human body occlusion and duplicated textures and limited power:

As for the smartphone-related software is considered, ADVIO dataset gives out all sorts of trouble that appear when you are doing indoor and outdoors path track by carrying around your iPhone or Google Pixel. VINS-Mono has some potential for an iPhone 14 Pro Max, they did some improvements on rolling-shutter artifacts problem as well in terms of corrections for sensor-IMU temporal and extrinsic calibration [20]. On the other hand, The IMU-Vision has been added onto a hand set and therefore serves as an accessible and expandable alternative that can also be used for human navigation. Commercial IMUs and rolling-shutter effects are reliable shortterms but not long terms [16].

In this paragraph, we give a collection of common methods that combine IMU-CAMERA and present them in Table II:

In summary, when it comes to complicated environment with temporary reliable and long time drift phenomenon, we suggest choosing IMU+vision FGO It's good at UAV and AR/VR stuff; has its benefits when the GNSS fails locally. But still regarding the Autonomous driving use case this is way more powerful on loop-closure and Multi-Session frameworks. In the smart phone and wearable devices it's practically not possible with retail grade sensors/synchronization.

2.3. IMU-LiDAR fusion via factor graph optimization

The fusion of IMU with LiDAR is becoming an essential technique to obtain precise SLAM and it is used in Autonomous vehicle, Mobile robot, complicated navigation etc. The LiDAR shape information and the IMU motion measurements get cleaned up around the same time in a factor graph sorta thing; when they join hands like that, everything gets taken care of nicely, no matter how speedy we are or how little there is to see for shapes. There are currently three categories of fusion architectures that roughly have a geometrically tight kind, filters+opt with optimisations and some prior info.

Geometric-tight frameworks, we are also taking care of LiDAR geometric error and IMU pre-integration Limitations in the Same optimization framework itself. Cai et al. [21] used a BEV-LIO(LC), which projected the LiDAR points into the BEV. This then deep network was utilized to derive the local context features (LCF's), and global context feature GCFs, by deriving the scan aligned with loops closed along with that of these. Using the FAST-LIO2 for the front end, alongside an iterative EK we carry our Back end of which it is a Factor graph optimization with odometry and loop closure. Empirical check on reduces absolute trajectory error (ATE) by 29% compared with FAST-LIO2. Setterfield et al [22].: bearing-range has something to do with an IMU limitation on the rotation LiDAR unit when it's got a high-maneuvering craft bound for landing And link everything back to the LiDAR through the IMU pre-integrated:

When we alter what is being scanned it changes how much its going to correct for movements but also has big increases when trying to land like on a planet mock up.

In terms of Hybrid filter-optimization strategies use filters in order to do a high frequency state estimate first and then they go on to use factor graph optimization so that there is global consistency. Tang et al. [23] presented LI Of LIO-FILO, where the LiDAR – IMU front end is based on iEKF, and the factor graph optimization back end is used as well as a light weight feedback loop to predict the next front- end state. On the UrbanLoco dataset trial shows that it gets to an absolute trajectory error of 0.202 m which is more than the state-of-the-art system like LIO-SAM and LeGO-LOAM. Zhou et al. [24] came up with an Indoor Location by using a Adaptive unscented filter along with factor graph optimization fo point-to-line lidar constraints the are different depeding if it has a strong feature match or no thought but still usin these. From what we see by observing the Indoor Corridor, It's very clear that using our suggested methods reduce localization error with 37% more than usual AUKF-ICP.

Degeneration resisting tech is focused on making the geom better or just putting a lot of constraints so as to have enough for an integration that's feature poor. Ai et al. [25] proposed a multi-line feature augmented LiDAR-IMU SLAM architecture, It's an iterated error-state Kalman Filter as front end and it uses the multi-factor graph optimization framework, this takes up on both LiDAR odometry data and IMU pre-integration data but does have limit with loop closure; In terms of testing with the KITTI dataset we get an RMSE of 2.17m and this is a 25% improvement over the filter only alternatives. And Ma et al. [26] put forward a multi-feature integration LiDAR - IMU SLAM and improved on the ICP-matching method by adding the pre-integration of the IMU to the point cloud motion correction. On M2DGR dataset it gets to an ATE RMSE of 0.197 m which is around 9.6% better in terms of precision compared to FAST-LIO2.

Table 3. IMU-LiDAR methods summary

Method	Ref.	Coupling	Core innovation	Calib.	Key result
BEV-LIO(LC)	[21]	LC	BEV representation + deep descriptors for scan matching & loop closure	No	29% ATE ↓ vs FAST-LIO2
Bearing-Range-IMU Factor	[22]	TC	Bearing-range-IMU factor for spinning LiDAR motion compensation	No	Improved accuracy in high-dynamic landing
LIO-FILO	[23]	Hybrid	iEKF front-end + FGO back-end with lightweight feedback	No	0.202 m ATE on UrbanLoco
AUKF-FGO	[24]	Hybrid	Adaptive UKF with point-to-line constraints + dynamic weighting	No	37% error ↓ vs AUKF-ICP
Multi-Line Feature Enhanced	[25]	Hybrid	Multi-line feature enhancement + multi-factor graph (LO, IMU, LC)	No	2.17 m ATE RMSE on KITTI (25% ↓)

Table 3. (continued)

Multi-Feature Fusion	[26]	TC	Improved ICP + IMU pre-integration for motion compensation	No	0.197 m ATE RMSE on M2DGR (9.6% ↓)
Map-Assisted SLAM	[27]	PA	OpenStreetMap structural constraints as auxiliary factors	No	Reduced urban drift
GNSS-Assisted SLAM	[28]	PA	GNSS factors aligned with LiDAR keyframes	No	33% accuracy ↑ vs ALOAM

For the existing knowledge bases, they can further help us improve on the reliability when localizing in longer term with the prior geographic auxiliary information. Ai et al. [27] They have applied guided Map with LiDAR-IMU SLAM through which they were trying to develop their created maps with reference from OpenStreetMap architectures by adding up all alignment constraint results as another constraints inside the factor graphs which resulted in lowering urban displacement. Lv et al. [28]: Give out the more improved GNSS-assisted LiDAR-IMU SLAM, which properly matches the correct GNSS measurements with LiDAR main pictures firstly. Then it takes these kinds of GNSS factors into account in its optimization framework On KITTI datasets show that this method has about 33% better localization than ALOAM.

In table III a synthesis of all the most typical IMU-LiDAR integrations studied in this section can be found:

In other words, an IMU-LiDAR factor graph SLAM is dependable for tough surrounding requiring swift dependability as it moves plus lengthen the paths that it corrects. It is above autonomous cars and drones; it can be very accurate for indoor automation, which helps in the reliability of many things outside. However it's still constrained by the hardware fidelity, processing power as well as what surrounds you.

2.4. IMU-4D RADAR fusion via factor graph optimization

Recently integrating 4D Radar with an IMU is turning out to be quite effective way of doing reliable motion tracking and positioning in complicated surroundings. There are also many works that combine radar Doppler information with inertial for better paths. 4D iRIOM framework takes in the radar inputs as well as move around a space obtained from an IMU do the entire act of radar- inertial odometry and map your way inside or outside [5]. Also persistent radar - inertial Dynamic Tracking was investigated in vast and very mobile cases to see how much better an accurate Predict would come for a Radar if it were given more Information via the use of inertial sensing [29]. The further works studied the application of the radar - IMU system on self driving cars, drones and different conditions for flying object under [30, 31]. Furthermore, there is the use of multiple sensor radar-IMU to help improve your ability with space and movement sense over more complex 3D terrain. More of this was further improved radar-inertial localization stability by modeling radar measurement uncertainty [32, 33].

Like this kind of radar-inertial combination have also tried in many places like Indoor Multi-story Buildings, Outdoor Citiescape and other wide settings. The joined up IMU data did better compared with the radar movement judgement as this gave quicker kinematics and had more coherence regarding the way. Using radar plus inertial sensors does improve accuracy of localization quite a

bit, it's stronger on all those different placement experiments [5]. Like when we mixed radar data with other information and it made finding things more exact, around 43%, in certain sets of numbers [29]. For the complex 3D landscape, I am using Radars + Inertials that lessens accumulation drift & has better trajectory stability [32].

The fusion of 4D radar plus IMU has various advantages regarding the motion estimation and localization. Radar device works normally under bad weather like rain, fog, dust, haze and bad visibility. In addition, Doppler signals provide instant velocity info that can be used to improve how well we estimate movements. But the data from radar is still little and messy, also its space information isn't clear enough. And there are other stuff like multipath reflections, ghost targets, weak vertical observability. You'll see it drift around through everything on all your moves. Therefore, we still need to do more work about optimizing our way of adopting 4D radar – IMU localization framework and its feasibility.

A summary of representative IMU-4D RADAR fusion methods discussed in this subsection is provided in Table IV.

Table 4. IMU-4D radar methods summary

Category	Content
Core Conclusion	4D radar + IMU fusion enables robust motion estimation and localization in complex environments.
Related Studies	4D iRIOM; continuous radar-inertial odometry; multi-radar configuration; uncertainty modeling.
Test Scenarios	Indoor multi-level, outdoor urban, large-scale dynamic scenes.
Key Results	Higher accuracy (~43% in some datasets); less drift; more stable trajectory.
Advantages	Radar works in bad weather; Doppler provides direct velocity; IMU offers high-frequency data.
Limitations	Sparse, noisy radar; low resolution; multipath, ghost targets, vertical observability issues.
Research Focus	Improve radar measurement usage and robustness for better localization.

3. Self-adaptive

3.1. Classical model-based adaptive strategies

As for traditional adaptives, the clear statistical framework is the answer; with an intuitive guideline it's easy; if we have clear signal stability metrics telling us when to change factors or network choice or selected variables. Such techniques are often thin, knowable, and do not need any training dataset, which is why they can be used immediately in many cases [34].

3.1.1. Threshold- and score-based adaptive weighting

The most used way is altering the measurement weights based on signal quality measurements or residual information. In terms of GNSS - IMU integration, RFGO [8] adjusts the Barron robust loss

function based on the pseudorange residual distribution. This will let them use heavier tailed stuff on probabilities in case some outliers showed up. Also makes it just a teeny tiny bit harder to be messed around by multipath or when you lose sight of something.

Radar-Imu Integration Adaptive technology uses things like dopler coherence and signal strengths as a kind of grade on whether to trust how often it sees something, also where that echo from the radar is reflected. The credibility of radar measurements will be quite variable due to multiple dispersed ways and things on the move. And Yang et al. [33] advise that it is necessary to consider variability between radar measurements, and coefficients values factors must be adjusted flexibly so as there would be good lasting consistency: Also Burnett et al. [29] found that the combination of Motion Anticipations and variable Radar prioritization greatly improves the Estimation Precision when there is high mobility. It's much like a complex multi-radar inertial odometry is even better at building things by coordinating info from lots of different types of radars [32]: In different situations selection of the data is also very important. Lyu et al. [30] recommend filtering out dynamic object-induced radar outliers so as to improve it very much. Like such a way, it's stressed that radar adaptation isn't only weighting, there's also data filter and choice.

And concerning LiDAR--IMU: the regular adaptive ones usually place importance on geometrical matching as well as centering at residuals [34]. A common way is what is done by Zhu et al. [35], this being an adaptive factor weighting approach, which judges the relative motion from LiDAR, vision, and IMU between consecutive key frames. An adaptive weight factor is added so as to flexibly adjust how much the LiDAR and visual stuff matters; when a sensor's idea of where it will go varies wildly from what the IMU predicted, that bit gets knocked down or thrown away completely. A little similar to the fashion which multiple sensor framework adaptations have been done through methods such ways used in MSF-SLAM [36] like these can also contain a variety of ways involving more complex layers where not only does the multi-modal loop closure take place but there's also hierarchical integration methodologies that do too.

3.1.2. Probabilistic and em-based uncertainty modeling

Heuristic weighting represents measurement uncertainty in a probabilistic manner as well. Much like the SLAC-FGO [7] EM with Gaussian mixture for residual giving it kind of ad hoc way to down weight unreliable data points that doesn't need any training.

This method gives basic directions on how to handle lots of different noise distributions that are commonly found in urban GNSS places. But those based on EM have some assumptions for residual distributions; it won't be right in extremely active or spread-out surroundings. And also mostly is related towards using just the pseudorange data, and it still quite complicated when trying to include more of these GNSS observations such as the carrier -phase signal which are still very challenging to have in adaptable architecture [9].

3.1.3. Structural adaptation in multi-sensor factor graphs

Apart from changing the metrics, restructuring the factor graph is also starting to emerge. These don't just mess with numbers; they change the actual graph structure themselves.

And then for example it could be like the best sliding window lengths that are kind of similar to how often you access these sensors. As for the ASW-FGO, it chose to handle adaptation by means of graphs and added more estimation time while there was no GNSS because that way they could still keep using their inertial as well as odometry sensors prior to going down again after returning with the signal [10]. Then we have even more complex ones based only off of an IMU and that is IFAL-

SLAM [37] where the authors adopt yet another dynamic sliding-window marginalization along with their other dynamic Lagrangian relaxations strategy that further enhances its stability performance on difficult tasks: For the same reason as multi sensors, will it be allowed or not to constraint when invalid? Even if it is not yet fully developed, structural adaption is still a good solution if the sensors are always worn or faulty.

3.2. Learning-based adaptive strategies

Data-driven adaptive methods mine latent nonlinear mappings between sensor observations and measurement uncertainty from large-scale datasets, and such methods may find those complicated, nonlinear links between sensing results and how true an observation is. As for current algorithms according to the particular work of the trained components: it is error variance predicting, intrinsic parameters adjusting as well as their coupling together with traditional factor graph refining.

3.2.1. Learning-based measurement uncertainty prediction

This group focuses on using neural architecture to do the calculation of the covariance or uncertainty of sensors directly, this would then be used as dynamic factors weights inside the factor graph:

With regard to the GNSS - IMU framework, Li et al. [6] used AFGO and applied a CNN to estimate the current position at any given time along with also changing the factor's covariance. It had nearly a 72% lower 2D RMSE for all types of receivers taught and this was great, however having to teach a receiver means it is limited to the training environment. Is different from a learnt uncertainty estimate which is able to pick out more reliable environment indicators.

In V_I Framework, there are some similar Ideas: Solodar and Klein [38]: VIO-DualProNet leverages neural networks to compute the IMU process noise covariances online thus giving us an opportunity of varying uncertainties. Makes a change from the old tough static-noise notion towards one which is based on figures, surroundings.

While it's very early to tell but learning driven uncertainty estimation is promising for a radar – IMU: Radar info is quite sparse and noisy, so it's tough to describe with just handmade features. Lately it's been on environment sensitive learning for finding mobiles removing false info, more than just regular stuff like: Deep Learning on Radar Quality Assessment, it is still a very promising one that's going to happen.

3.2.2. Learning-based self-calibration and its support for adaptive fusion

Other than forecasting about some uncertainty, I do believe it would be interesting if we were able to also use learning for evaluating an intrinsic or extrinsic sensor variable and have the system do this even on demand so it can stay true when integrating temporally.

It is a main focus in the field of visual-inertial. Qin and Shen [39] - Real-time temporal calibration has found that the cam-IMU sync has gaps when optimizing, and better result can be gotten if there are much different between IMUs. Extending this idea, Yang et al. [40] proposed a unified self-calibration framework to measure both intrinsic value of spatial–temporal relation and rolling shutter in order to get overall measurement.

Such work shows going from fixed cal and static noise to ongoing adjustment and updating of the uncertainty estimate, which is still key for the mass market versions operating all the time. It's continually improved on as sensors improve, this is the indirect way of maintaining our fuses coefficient real so we have an initial stage.

3.2.3. Integration with classical optimization frameworks

One important feature about the aforementioned learn to refine strategies is its nice fusion with traditional factor graph refinements. The forecast is not replaced but combined with the learned forecasts regardless of uncertainty assessments or calibration coefficients to be incorporated into the information matrix or factors of FGO as shown in both GNSS [6] and visual–inertial systems [38].

So such mixed systems can get truths from statistic and potential of a neuro, so give same answer in different places: The close combination of various kinds of training component with different optimization engines is an important part of modern study on multi-sensor integration.

4. Discussion and conclusion

In this article, we look at an IMU based Multi-Sensor Integration under Factor Graph Optimization architecture which includes the IMU-GNSS, IMU-Camera, IMU-LiDAR-Camera and IMU-4D Radar configuration. The Adaptive Weighted Fusion(AWF) is taken as one of the main techniques. For every case in the 4 modality: fgo is always better or on par with any filter method using all measurements and considering everything over a large trajectory window, yet still keeping it possible to have individual weightings per measurement. Thus this built in structure is also what makes it natural that AWF would be next: the framework inherently allowed for individual factors weights, the main thing was that those weightings needed to be trusted and flexible.

1. Noise modeling matters more than graph topology. IMU-GNSS: Looking at what I think is a way to improve our noise model by doing some kind of robust loss or probabilistic mixture and predicting all uncertainties, it does better than any change made in anything lower in the graphs. And this probably holds true for different modalities as well - impaired LiDAR features and radar multipaths are just two flavors of noisy measurements, and progress there will go down the same path.

2. AWF is disjointed from different kinds of sensors. Almost all current adaptive techniques were made for only one type of modality and do not work when moving to another sensor. I don't see any architecture implementing AWF by uniformly using it in one single factor graph, three or more. With multisensor setups as the new normal, think of the IMU+CAMERA+LiDAR+GNSS framework, it is this gap that matters most. It is still not a framework seeing that AWF is an important fundamental goal instead of just a per-sensor module.

3. High-precision observables remain outside adaptive frameworks. While millimeter-level accuracies can be achieved from carrier phase GNSS data, but it has some tricky integer ambiguity pattern problem that today's adaptive noise models cannot cope with. It is observed that similar to sparse issues in radar point cloud, we also can't train the continual-learning-weighting techniques. Of all the best metrics on each modality is also where there will be the least amount of adaptive tools and that is the max possible precision for modern systems.

Looking forward, there seem to be 3 paths ahead: At the start when it comes to interacting with cross-modal uncertainty propagation which takes use of number in the quality 1 sensor to calibrate how prioritizing another sensor be done actively rather passive through adaptive mix. Second is structure of the graph itself rather than going over each factor and changing its weight for reaction to sensor malfunction or deterioration. As far as how to move a research prototype along into being embedded live, this is another area where there's not great existing analysis -- and what does so often entail a bit more than just some computational chops; making things transparently clear also counts.

FGO +AWF is a good base of technical things for more advanced navigation frameworks. Factor graphs with their clarity on each measurement's contribution--- separate from changes, they do particularly good around all sorts of different moving and shifting sensors in that wild world. This is a review of what I have looked up, stating where the discipline currently stands, and it points very specifically to which important future advancements are yet to be accomplished.

References

- [1] Thrun, S., Burgard, W., & Fox, D. (2005). *Probabilistic robotics*. MIT Press.
- [2] Dellaert, F., & Kaess, M. (2017). Factor graphs for robot perception. *Foundations and Trends in Robotics*.
- [3] Barfoot, T. D. (2017). *State estimation for robotics*. Cambridge University Press.
- [4] Groves, P. D. (2013). *Principles of GNSS, inertial, and multisensor integrated navigation systems*.
- [5] Zhuang, Y., Wang, B., Huai, J., & Li, M. (2023). 4D iRIOM: 4D imaging radar inertial odometry and mapping. *IEEE Robotics and Automation Letters*.
- [6] Li, Z., et al. (2024). Intelligent environment-adaptive GNSS/INS integrated positioning with factor graph optimization. *Remote Sensing*, 16(1), 181.
- [7] Bai, X., et al. (2023). An enhanced adaptable factor graph for simultaneous localization and calibration in GNSS/IMU/odometer integration. *IEEE Transactions on Vehicular Technology*, 72(6), 7430–7444.
- [8] Ahmadi, E., et al. (2025). Adaptive factor graph-based tightly coupled GNSS/IMU fusion for robust positioning. *IEEE Transactions on Vehicular Technology*.
- [9] Beuchert, J., Camurri, M., & Fallon, M. (2023). Factor graph fusion of raw GNSS sensing with IMU and lidar for precise robot localization without a base station. *arXiv:2209.14649*.
- [10] Ji, X., et al. (2025). GNSS/IMU/ODO integrated navigation method based on adaptive sliding window factor graph. *Electronics*, 14(1), 124.
- [11] Forster, C., Carlone, L., Dellaert, F., & Scaramuzza, D. (n.d.). On-manifold preintegration for real-time visual-inertial odometry. *arXiv*.
- [12] Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R., & Furgale, P. (n.d.). Keyframe-based visual-inertial odometry using nonlinear optimization.
- [13] Qin, T., Li, P., & Shen, S. (n.d.). VINS-Mono: A robust and versatile monocular visual-inertial state estimator. *arXiv*.
- [14] Campos, C., Elvira, R., Gómez Rodríguez, J. J., Montiel, J. M. M., & Tardós, J. D. (n.d.). ORB-SLAM3: An accurate open-source library for visual, visual-inertial and multi-map SLAM. *arXiv*.
- [15] Rosinol, A., Abate, M., Chang, Y., & Carlone, L. (n.d.). Kimera: An open-source library for real-time metric-semantic localization and mapping. *arXiv*.
- [16] Cortés, S., Solin, A., Rahtu, E., & Kannala, J. (n.d.). ADVIO: An authentic dataset for visual-inertial odometry. *arXiv*.
- [17] Wenzel, P., Wang, R., Casser, V., et al. (n.d.). 4Seasons: A cross-season dataset for multi-weather SLAM in autonomous driving. *arXiv*.
- [18] Burri, M., Nikolic, J., Gohl, P., Schneider, T., Rehder, J., Omari, S., Achtelik, M. W., & Siegwart, R. (n.d.). The EuRoC micro aerial vehicle datasets. ETHZ ASL.
- [19] Schubert, D., Goll, T., Demmel, N., Usenko, V., Stückler, J., & Cremers, D. (n.d.). The TUM VI benchmark for evaluating visual-inertial odometry. *arXiv*.
- [20] Ling, Y., Bao, L., Jie, Z., Zhu, F., Li, Z., Tang, S., Liu, Y., Liu, W., & Zhang, T. (n.d.). Modeling varying camera-IMU time offset in optimization-based visual-inertial odometry.
- [21] Cai, Y., Xu, J., & Zhang, S. (2023). BEV-LIO(LC): LiDAR-inertial odometry and loop closure based on bird's-eye-view representation. *IEEE Robotics and Automation Letters*, 8(7), 4142–4149.
- [22] Setterfield, T., Lambert, A., & Sukkarieh, S. (2023). A tightly coupled LiDAR–inertial navigation system for high-dynamic motion. *IEEE Transactions on Aerospace and Electronic Systems*, 59(4), 3412–3425.
- [23] Tang, J., Li, X., & Kaess, M. (2024). LIO-FILO: LiDAR-inertial odometry with filter-based front-end and factor graph optimization. *IEEE Robotics and Automation Letters*, 9(2), 1456–1463.
- [24] Zhou, H., Chen, Y., & Li, Z. (2023). Adaptive unscented Kalman filter and factor graph optimization for indoor LiDAR-IMU localization. *Sensors*, 23(18).
- [25] Ai, X., Wang, H., & Liu, Y. (2024). Multi-line feature enhanced LiDAR–IMU SLAM for robust localization. *IEEE Access*, 12, 33245–33258.

- [26] Ma, Y., Zhang, Q., & Liu, J. (2025). Multi-feature fusion LiDAR–IMU SLAM based on factor graph optimization. *IEEE Transactions on Instrumentation and Measurement*, 74.
- [27] Ai, X., Wang, H., & Liu, Y. (2024). Map-assisted LiDAR–inertial SLAM with structural constraints. *Remote Sensing*, 16(3).
- [28] Lv, C., Deng, Z., & Zhao, H. (2024). GNSS-assisted LiDAR–inertial SLAM for accurate localization in urban environments. *IEEE Sensors Journal*, 24(6), 9123–9134.
- [29] Burnett, K., Schoellig, A. P., & Barfoot, T. D. (2024). Continuous-time radar-inertial and lidar-inertial odometry using a Gaussian process motion prior.
- [30] Lyu, Y., Hua, L., Wu, J., Liang, X., & Zhao, C. (2024). Robust radar inertial odometry in dynamic 3D environments. *Drones*.
- [31] Coto-Elena, L., Caballero, F., & Merino, L. (2025). Loosely coupled 4D radar-inertial odometry for ground robots. *Journal of Intelligent & Robotic Systems*.
- [32] Huang, J.-T., Xu, R., Hinduja, A., & Kaess, M. (2024). Multi-radar inertial odometry for 3D state estimation using mmWave imaging radar.
- [33] Yang, W., Jang, H., & Kim, A. (2025). Ground-optimized 4D radar-inertial odometry via continuous velocity integration using Gaussian process.
- [34] Fan, Z., Zhang, L., Wang, X., Shen, Y., & Deng, F. (2025). LiDAR, IMU, and camera fusion for SLAM: A systematic review. *Artificial Intelligence Review*, 58(6), 174.
- [35] Zhu, Z., Zhang, Y., Wang, W., Feng, W., Luo, H., & Zhang, Y. (2023). Adaptive adjustment of factor's weight for a multi-sensor SLAM. *Journal of Physics: Conference Series*, 2451(1), 012004.
- [36] Lv, X., et al. (2024). MSF-SLAM: Multi-sensor-fusion-based SLAM for complex dynamic environments. *IEEE Transactions on Intelligent Transportation Systems*, 25(12), 19700–19716.
- [37] Liu, J., Qi, Y., Liu, L., Zhou, Y., & Zhang, Y. (2025). IFAL-SLAM: Inertial-centered multi-sensor fusion with adaptive Lagrangian optimization. *Measurement Science and Technology*, 36(1), 016336.
- [38] Solodar, D., & Klein, I. (2024). VIO-DualProNet: Visual-inertial odometry with learning based process noise covariance. *Engineering Applications of Artificial Intelligence*.
- [39] Qin, T., & Shen, S. (2018). Online temporal calibration for monocular visual-inertial systems. *IROS*.
- [40] Yang, Y., Geneva, P., Zuo, X., & Huang, G. (2023). Online self-calibration for visual-inertial navigation: Models, analysis, and degeneracy. *IEEE Transactions on Robotics*.