

Graphic Visualization of Probabilistic Traffic/Trajectory Predictions in Mobile Applications. A First Prototype and Evaluations for General Aviation Purposes

Giuseppe Frau^{1,3(✉)}, Francesca De Crescenzo²,
and Damiano Taurino³

¹ Faculty of Engineering, Roma III University,
Via della Vasca Navale 79/81, Rome, Italy
giuseppe.frau@dblue.it

² Department of Industrial Engineering, University of Bologna,
Via Zamboni 33, Bologna, Italy
francesca.decrescenzo@unibo.it

³ Deep Blue srl, Piazza Buenos Aires 20, Rome, Italy
damiano.taurino@dblue.it

Abstract. The present work describes the interactive prototype and the preliminary evaluation results of a tool dedicated to the light General Aviation pilot's community. The tool's interface has been developed through an Android tablet application and aims at supporting the pilots in the task of staying "well-clear" from the surrounding traffic by presenting them the long-term prediction of the flights. The initial results and the approach of a heuristic evaluation conducted with five experts coming from the fields of user-experience, aviation and automotive are discussed along with the improvements in the design of the user-interface focusing on the trajectory depictions.

Keywords: User-interface design · Heuristic evaluation · Light general aviation · Trajectory prediction visualization

1 Introduction

Light General Aviation (Light GA) pilots represent a little known category of users compared to the commercial ones. These pilots fly small aircraft in uncontrolled airspaces and rely on visual means to navigate and to keep themselves separated from the surrounding traffic. When navigating by visual means, Visual Flight Rules (VFR) are applied to operate the aircraft. Moreover, pilots flying under VFR conditions and in uncontrolled airspaces are not bound to submit any flight plan to any sort of authority. By contrast, in the Commercial Aviation segment, Instrumental Flight Rules (IFR) are applied and flights are operated in controlled airspaces for which formal flight plans are submitted.

Navigation and separation by visual means, however, have several limitations that may lower the safety of the flight [1]. Between 2009 and 2013, for example, EASA reported a number of 30 mid-air collisions for the Light GA segment within the

European airspaces [2]. As a comparison, in the Commercial Aviation segment, a number of 4 mid-air collisions have been registered between 2004 and 2013 [2]. Such data solicit the need for more powerful decision support tools to provide information and interact with data that can increase the pilot situation awareness (SA). Concerning the projection component of SA a major issue is in designing tools that can effectively support pilots in taking a safe decision. In this context they rely on projections built upon their experience and their knowledge of the traffic they fly in.

The purposes of this paper are (a) to present a prototype tool dedicated to light GA pilots and developed within the ProGA¹ project aiming at improving their traffic awareness and their safety by the provision of long-term predictions of the surrounding flights, and (b) to show the initial results of the evaluation phase.

The first part of the paper is dedicated to the description of the prototype. After a general introduction is given, the concept at the base of the algorithm for the long-term trajectory prediction is briefly outlined. The following sections are devoted to the user-interface description.

Section 3 defines the method used for the evaluation along with its rationale. Finally, Sect. 4 shows the results of such approach, while conclusions are drawn in the last of the work.

Prediction Algorithm Concept. This section presents the concept on which the trajectory prediction algorithm of the ProGA system is based. This algorithm is the main source of information of the prototype evaluated in this work but it is worth to underline that its functioning has to be considered out of the scope of the evaluation.

The kinematic state (position, speed, turning rate, etc.) of an aircraft determines the so-called state-based prediction, which is obtained by plugging the kinematic state into a dynamical model of the aircraft motion. State-based predictions are reliable over a short timeframe (less than 30 s) but may prove more and more untrustworthy if casted over an increasing time interval. This behavior is in fact inherent to all kind of predictions.

The ability of casting reliable long-term predictions (2 to 5 min ahead) is an asset for ProGA because it is expected to likely increase the situational awareness and safety of the user. For this reason ProGA introduces intent-based predictions, which determine the future trajectory of the aircraft exploiting the notion of flight intent, i.e. the sequence of straight legs that the pilot has planned to fly; see [3] for further details.

Flight intents may be shared by pilots prior to takeoff or may be statistically inferred from a (local) base of knowledge describing recurring patterns and behaviors common to GA flights operating in a certain area and under specific conditions. ProGA considers those two classes of inputs, i.e. shared flight intents and inferred ones, as complementary to each other. The prediction algorithm works in fact with a subset of flight intents picked up from the union of the declared intent, if any, and the most statistically significant recurring paths. The final prediction is carried out by means of Monte Carlo simulations of the aircraft motion along the intents selected as above. The interested reader is referred to [3] for further information about the ProGA prediction algorithm.

¹ Probabilistic 4D Trajectories of Light General Aviation Operations www.proga.nlr.nl.

2 Prototype Description

The present section illustrates the object of evaluation of the present work, namely the user interface layer of the tablet application developed for supporting the pilot in the usage of ProGA during the in-flight phase.

The ProGA in-flight interface consists of a tablet application developed using the Android 5.0 SDK. The application is built around five main components:

- A moving map constantly centered on the own-ship position and oriented towards the own-ship speed direction;
- The own-ship and the surrounding traffic representations;
- The traffic information label;
- The global functions menu;
- The prediction visualizations.

The tablet application represents the layer for human interaction of a more articulated prototype, which consists of other two blocks, namely the traffic simulator (in which the prediction algorithm is also plugged) and the flight simulator for future tests with real pilots.

Moving Map. The moving map component represents the main view of the application. Its purpose is to provide awareness to the pilot about his/her geographic position and, at the same time, to work as a container of the surrounding traffic representation and of the predicted trajectories. The map component is provided by the Android SDK and it uses the Google Maps v2 APIs that allows the drawing of 2D primitives on the map (functionality that has been used for drawing the predicted trajectories). Figure 6 shows the main view of the application including the moving map component, the traffic representation and the global functions menu. The ProGA moving map follows the behavior of the standard moving maps to which most of the people are by now used to. As such it responds to the standard gestures such as pinch-to-zoom, pan-to-move etc.




Own-ship and Traffic Representation. All the aircraft in the map are depicted using a colored arrow with a white border. Since the background behind the arrows is dependent on the current area -hence the colors are not predictable- the white border assures a minimum level of contrast also in the situations in which the color of the arrow is similar to the color of the current background.

The aircraft positions are updated at the same frequency of the map and of the own-ship ones. Moreover, while the color of the own-ship arrow is constantly black, the traffic arrows follow a color-coding scheme that depends on two factors, namely (Table 1).

1. The difference of altitude between the own-ship and the specific aircraft;
2. The result of the “Monitor” functionality for that specific aircraft;

Traffic Information Label. While the information about the position of the surrounding traffic is naturally conveyed by the depiction on the map relatively to the own-ship one, there are a set of other information—namely the aircraft identification number, and the altitude- in which the pilot may be interested and that is displayed on demand in order to avoid cluttering on the map view.

Table 1. Color-coding schema for traffic representation

Condition	ABS(own-ship alt – aircraft alt) \geq 500 ft.	ABS(own-ship alt – aircraft alt) \leq 500 ft.	Dangerous
Representation			

In order to obtain these information, the pilot has to interact with the traffic arrow by a simple tap and, in this way, opening the so called Information label which consists of a floating box placed nearby the selected aircraft. Through the information label, the pilot can accomplish two goals:

1. Obtaining altitude and identification number about the selected aircraft.
2. Access to the prediction request commands: a set of buttons that allow the pilot to enquiry the system about the long-term prediction of the selected aircraft.

This double goal is reflected by the two possible modes that the information label is able to assume. In the compact mode, the label only shows the additional information of the aircraft i.e. altitude and identification number. In the expanded mode, the label size is augmented to host the prediction request buttons (Fig. 1).

The identification number of the aircraft (in these cases “NINJA34”) is reported on the top left of the floating box, while the altitude (2000 feet) is on the top right. Each time the label is tapped, it switches from the compact mode to the expanded one and vice versa. The label can also be totally hidden by tapping outside of the box anywhere on the map view.

As previously mentioned, the information label also hosts the prediction request buttons. In the worst case, for requesting a prediction, the user needs to tap once on the target aircraft, then expand the label, and finally select the desired prediction time horizon. The system offers four possible time horizons for the prediction: from 2 min to 5 min with steps of 1 min. However, when the user selects a time horizon, the system doesn't merely ask for that time horizon only. A prediction request of 5 min would generate a cloud of point far enough to be isolated from the aircraft that generated it. For this reason, the system splits each request in a set of multiple requests of incremental time horizons until the target one is reached. For example a 4 min prediction request by the user generates a set of requests for predictions at 1 min, 2 min, 3 min and finally 4 min in the future. This allows the system to build an actual trajectory (or corridor) by merging the resulting data.

Prediction Visualizations. Within the developed prototype two main classes of data coming from the central system can be identified:

- Corridors data: they can be considered as the raw data produced by the system. When a prediction for a given aircraft (indicated by the pilot) is requested, the

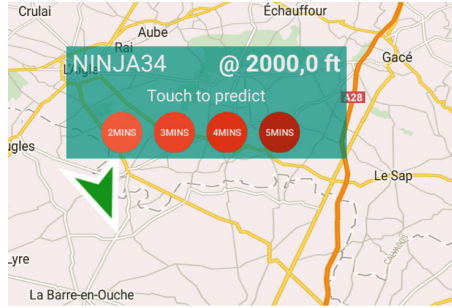


Fig. 1. Traffic information label expanded mode

prediction algorithm produces a cloud of possible points in which that aircraft is likely to be in the future;

- Elaborated data: pilots can also enquire the system through the Hotspots tool and through the Monitor too. This two functionalities still use the same prediction capabilities but that add a processing stage to the resulting data aiming at finding more punctual information i.e. the hotpots position and the potential intruders.

The major efforts in the design and development of the visualization have been mainly directed to the first class of data. Three alternative depictions are implemented in the prototype. The following subsections explain all the depiction modes and the rationale behind them.

Note that the design of these depiction had to take into account the limitation of computing capacity in the mobile device used for the realization of the prototype. In fact, although the mobile device has been chosen inside a range of current top devices, the amount of data received and the constant need to update the whole presented scenario, still require more powerful computation resources. The following subsections illustrate the three alternative depiction modes designed for the prototype.

Particles Visualization Mode. The particles visualization is an attempt to depict the prediction results without adding any extra-processing or information layer to it. Each point inside the cloud coming from the prediction algorithm is considered as a singular entity –the particle- carrying no other information but its position. Figure 2 presents a screenshot in which a prediction of 5 min in the future for the aircraft placed on the right of the map was requested. As previously mentioned, data about 4, 3, 2 and 1 min in the future is also requested when the system is enquired; considering the full set of received data, some emergent properties can be deduced from the depiction:

- Continuity: the particles tend to naturally form a corridor if the prediction is sharp enough for the target aircraft.
- Probability Conveyance: each particle is assigned with a constant size and level of transparency [4, 5], therefore, the formed corridor tend to be more opaque when more particles overlap (even partially) at the same location, hypothetically conveying a concept of higher-probability for these parts of the corridor.



Fig. 2. Particles visualization mode

These two properties, however, are not verified in the following circumstance: the ProGA system defines a set of parameters useful to tune the algorithm behavior. The number of particles used to cast the predictions is defined by one of these parameters. A high number of particles are advisable in order to have a good quality of the results. However, when brought on a mobile device, the number of data points can't be too large due to the computational limitations. If the chosen number of particles used to cast the prediction is too low, the emerging corridor turns into a series of circular clouds of points breaking the both the described properties.

Heatmap Visualization Mode. The heatmap visualization aims at representing the natural evolution of the Particles one and it is designed to make explicit the two emergent properties of continuity and probability conveyance. In this implementation, the locations with higher probability of being occupied by the target aircraft in the selected time frame are represented with a shade of red, while as the probability lowers the hue associated tends to the color green. Hypothetically, heatmap visualizations should naturally convey the density of the data-points and at the same time provide an idea of the trajectory evolution when the uncertainty in the result is not high enough to uniformly spread the points in the different directions. Figure 3 shows a screenshot of a 5 min prediction casted for the aircraft on the right side of the image.

Connected Particles Visualization Mode Finally, the connected particles depiction aims at overcoming the limitations of the Particle one when used with a limited set of data points and at the same time is directed to provide a sharper view of the trajectory. To accomplish these objectives, this design introduces some variants to the Particle visualization mode (Fig. 4):

- Particles representation is sensibly bigger in order to have overlaps circles even when fewer particles are presented on the map.
- The particles of each prediction group are connected with the particles of the previous and the next group. For example, let's consider a situation in which a

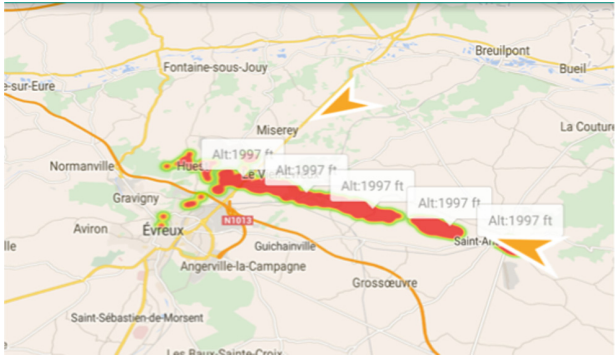


Fig. 3. Heatmap visualization mode



Fig. 4. Connected particles visualization mode

3 min prediction is requested. This prediction is broken in three sub-predictions of 1, 2 and finally 3 min. At this point, all the particles resulting from the 1-min. predictions get connected by a line with the respective particle of the 2 min group.

3 Evaluation Approach

In the previous sections, a prototype for the light GA pilots has been described, both under its functioning and human interaction sides. The present section illustrates the approach that has been followed in the initial evaluation of the prototype. Being the prototype on its early maturity level, the choice of the approach has been the one of the heuristic evaluation with experts [7]. However, since the application context represents a challenging factor -due to its criticalities and domain complexity-, an accurate choice of the evaluators was required. In particular the selection criteria was such to maximize the number of expertise fields within the all set of evaluators. Five experts were selected for the analysis. The fields of expertise for each evaluator are summarized in the Table 2.

Table 2. Expertise fields of the selected evaluators

Evaluator ID	Fields of expertise
Evaluator 1	Automotive, Human Factors, Aviation
Evaluator 2	Human Factors, Aviation
Evaluator 3	User-Experience, Aviation Safety, Patient Safety
Evaluator 4	User-Experience, Usability, Healthcare, Aviation
Evaluator 5	Usability, Automotive, Aviation

After being introduced to the concepts of the prototype, each evaluator has been individually asked to inspect the application, explore it and ask questions when needed. Evaluators were then asked to provide their feedbacks about the prototype. In particular feedbacks on the usability issues and on the quality of the trajectory prediction visualizations were stressed. In order to justify every found issue, the evaluators were asked to find possible consequences due to the presence of the identified issue.

The issues were, at this point, aggregated categorized and ranked. Categorization is based on which of the famous Nielsen's heuristics [6] they were found to go against. The ranking of each issue, conversely, followed a two-step process.

- (a) The visibility index " V " was calculated, namely, how many evaluators detected the issue;
- (b) The impact index " I " was decided on a scale between 1 and 3 where the highest value was associated to the issues directly impacting on the main user's goal, 2 was assigned to issues indirectly impacting the main goal or impacting secondary functions; finally, the value of I was assigned to all the other issues.

The final was obtained through the application of the following formula.

$$r = (I \times n) + V$$

Where n is the number of evaluators. The obtained raking has the property of giving more importance to the issues with the highest impact, and at the same time generating a local impact order based on the visibility, ergo the issues with the same impact are ordered by their visibility value.

4 Results of the Evaluation

The evaluation with the experts highlighted a number of 28 unique issues in the prototyped user interface. Each evaluator found on average 10 issues (not unique) during the review of the prototype. Looking at the unique issues, the trend shown in Fig. 5 was found to be followed in their identification, while the distribution of the issues among the not-respected Nielsen's heuristics is presented in Fig. 6.

The trend shows a growing number of overall issues that is likely to reach a plateau after the fifth expert [7], while the plateau for the important ones is reached in the very early stages of the evaluation.

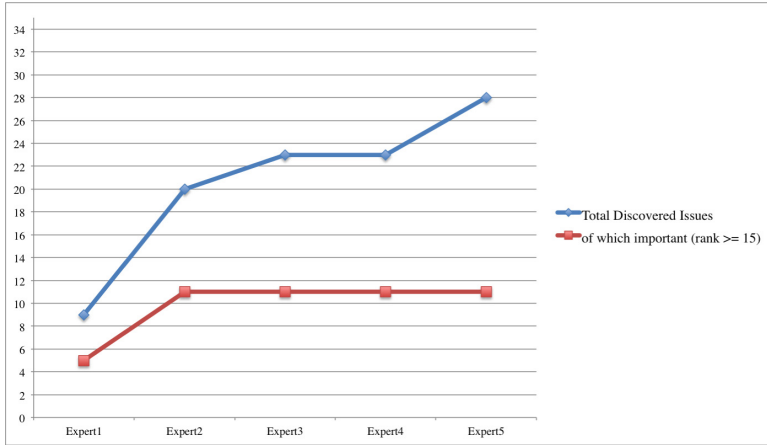


Fig. 5. Discovered issues after each expert evaluation. The upper trend shows the cumulative number of unique issues reached after the evaluation of each expert. The bottom one, instead, outlines the cumulative number of those issues considered important taking into account the original rank scale ($r \geq 15$ on a range from 1 to 20).

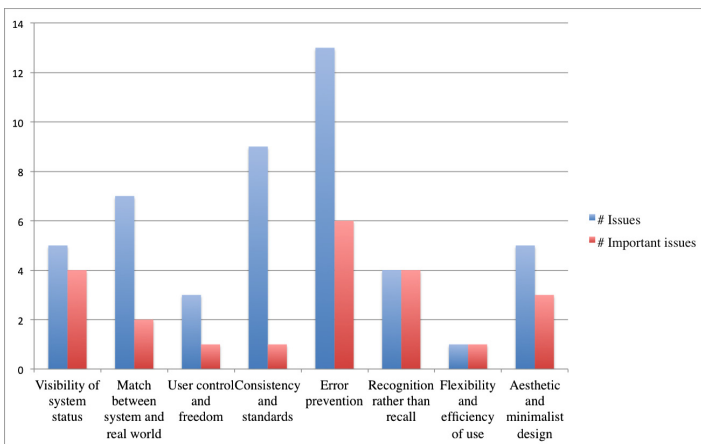


Fig. 6. Distribution of the issues among the Nielsen’s heuristics

In the issues distribution mainly two aspects can be noticed: from one hand, the found issues mainly targeted the heuristic regarding “*Error prevention*”, “*Consistency and standards*”, “*Match between system and real world*”, “*Visibility of system status*” and “*Aesthetic and minimalist design*”. From the other hand, when looking at the portion of the issues considered important by the developed ranking method, a sensible flattening of some heuristics -that we can now consider as categories for the issues- can be appreciated. For instance, the “*Consistency and standards*” heuristic loses out 8 points passing from the total view to the important issues one, while categories like

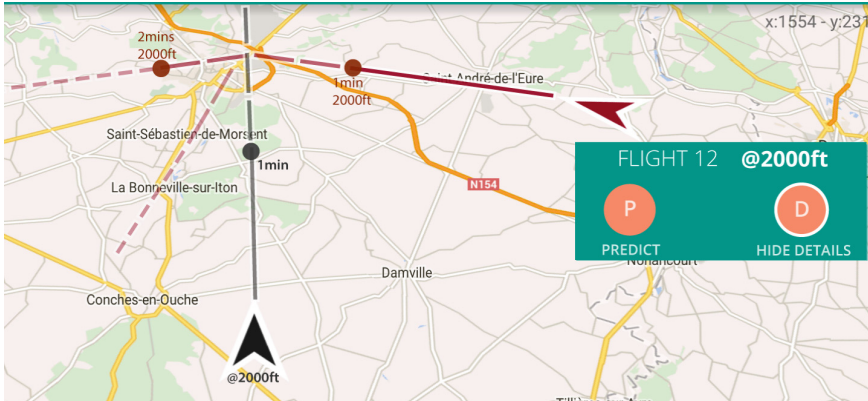


Fig. 7. Proposal of redesign

“*Visibility of system status*” and “*Recognition rather than recall*” do not experience such a sensible reduction.

Finally, considering output of the experts’ analysis and in particular the most important issues, a redesign of the prediction visualization was produced and it is here presented (Fig. 7).

The redesign aims at solving the following identified top-issues:

- (1) Intrusiveness of the prediction visualizations, causing cluttering on the screen and possible confusion with consequent disregard of the main function of the tool;
- (2) Lack of the temporal dimension indication on the trajectory: causing difficulties or the even the impossibility for the user to interpret the output and possibly leading to mistakes in the decision making process;
- (3) Lack of the own-ship prediction: in order to properly understand whether the surrounding traffic is going to cross the own-ship trajectory, the indication of such information must be presented for comparison purposes and it is considered a fundamental support for the main goal of the tool.

The re-design aims at solving these issues by providing a graphical notation of the time dimension, which is represented in a consistent way both on the intruder prediction and on the own-ship one. Moreover, in order to avoid cluttering, only the trajectory with highest probability is presented while the alternative ones are displayed on demand.

5 Conclusions

This work presented the prototype of a tool for GA pilots aiming at supporting the task of “*staying well clear*” from the surrounding traffic. In addition, a first heuristic evaluation was conducted.

From the design and development point of view several challenges had to be addressed: long-term prediction (from 2 to 6 min) is a difficult task when applied to

light GA flights. The result of this prediction is of probabilistic nature and carries a high quantity of uncertainty, which is expected to grow as the look-ahead of the prediction goes farther. As a consequence, in order to exploit this kind of data we need to design a visualization framework capable of communicating both the potentially useful information for the pilot to make his decisions and its level of uncertainty. The main components of the user interface have been described in the first part of the work with a focus on the prediction visualizations and their rationale.

The heuristic evaluation of this interface was conducted following an approach that contemplated the selection of five evaluators based on their expertise fields and the analysis of the found issues based on a ranking method that allowed us to inform the subsequent re-design proposal. Results of this work will be used to better understand the impact of each design parameter on the overall efficiency and usability of the interface. Moreover, the design of the interactive display for general aviation will be further refined. This is one of the first attempts to introduce the visualization of uncertain data to support the human in the estimation of the projection of events in the operational environment of general aviation. Except the meteorological data, which are the uncertainty that the pilots are used to deal with, the evolution of the general aviation traffic is another component that needs to be introduced in the cockpits HCIs in order to increase the safety.

References

1. Graham, Walton, Orr, Robert H.: Separation of air traffic by visual means: an estimate of the effectiveness of the see-and-avoid doctrine. *Proc. IEEE* **58**(3), 337–361 (1970)
2. EASA Annual Safety Review 2013. <http://easa.europa.eu/newsroom-and-events/general-publications/annual-safety-review-2013>
3. Lancia, C., et al.: Traffic predictions supporting general aviation. In: Fourth SESAR Innovation Days Conference
4. Bertin, J., Berg, W.J.: *Semiology of Graphics: Diagrams, Networks, Maps*. University of Wisconsin, Madison (1983)
5. MacEachren, A.M., et al.: Visualizing geospatial information uncertainty: What we know and what we need to know. *Cartography Geographic Inf. Sci.* **32**(3), 139–160 (2005)
6. Nielsen, J.: Enhancing the explanatory power of usability heuristics. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM (1994)
7. Nielsen, J., Rolf, M.: Heuristic evaluation of user interfaces. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM (1990)