Abstract—We use motion context recognition to enhance the result of our infrastructure-free indoor navigation algorithm. Target applications are difficult navigation scenarios such as first responder, rescue, and tactical applications. Our navigation algorithm uses inertial navigation and visual navigation fusion. Random Forest classifier algorithm is taught with training data from Inertial Measurement Unit and visual navigation data to classify between walking, running and climbing. This information is used both in pedestrian navigation to do stationarity detection with adaptive threshold and in particle filter fusion to exclude visual data from during climbing. Methods are evaluated in an indoor navigation test where person wearing tactical equipment moves through a building. Results show improvement of positioning accuracy based on loop closure error on the test track especially when the movement is fast paced. The loop closure error was reduced on average 4% in two tests when movement was slow and 14% when movement was fast.

Index Terms—Context awareness, Indoor navigation, Inertial navigation, Sensor Fusion, Visual odometry

I. INTRODUCTION

Navigation systems that work in any location are especially useful in for example first responder, rescue, and tactical applications, where pre-existing navigation infrastructure cannot be expected to be present indoors. Our research attempts to develop infrastructure-free indoor navigation and situational awareness system [1]. Freedom from infrastructure consists of using only self-contained sensors and not preinstalled equipment, such as Wi-Fi and Bluetooth positioning. Preliminary knowledge of the location, such as floorplans or maps, are not used either in infrastructure-free navigation. Developing an infrastructure-free navigation system is a challenging task but makes it possible for the system to be used anywhere.

Self-contained sensors include Inertial Navigation Systems (INS) that use Inertial Measurement Units (IMUs) to measure acceleration and turn rate. Camera is also a self-contained sensor and can be used in indoor navigation to obtain turn rate and translation estimates visually [2]. While cameras and INS can be used anywhere and require no infrastructure, they do have a limitation that they can only estimate relative position. Visual navigation and INS require a known starting location and their result will inevitably drift away from the true location as time passes from the last known location update. For this reason it is necessary to use every possible method to increase the time the sensors can give accurate position estimate.

Situational awareness in navigation consists of all the information of the user’s surrounding environment and conditions in addition to their location [3]. This can also be called the navigation context. Context information is valuable in itself but the information about the user’s context can also be used to improve the navigation solution. Navigation systems always operate in some context they are built for and many modern systems already navigate in a number of different contexts [4]. For example a smart phone that can be carried on a pedestrian or on a car, etc. [5]. A context adaptive system should be able to recognize this change automatically.

It is difficult to develop an algorithm that is suited for variable conditions especially in applications such as rescue and tactical operations. Instead, it may be useful to change the algorithm based on situation, for example based on how the user moves [6]. Some examples of this context adaptive navigation include works such as [7] where the navigation system seamlessly transfers between indoor and outdoor positioning systems when necessary. In [8] the authors detect features in indoor environments such as elevators, escalators and stairs through pedestrian activity recognition. This information is used to enhance pedestrian dead reckoning.

In this study we recognize the motion context using machine learning and use this knowledge to determine whether the user is running, walking, or climbing. Several studies have shown that it is possible to detect the motion context of a pedestrian – in other words, how the user moves – from sensor measurements, for example [1], [9]. This information was also recognized and applied to inertial navigation algorithms in [10], [11]. We use machine learning to create a pedestrian navigation algorithm that can recognize contexts walking, running, and climbing and adapt based on the detected context. We then fuse the results with visual navigation data in particle filter presented in [12]. For the particle filter we use the visual data based on context: during climbing the visual data will not be used.

This work was supported by the Scientific Advisory Board for Defence of the Finnish Ministry of Defence [project INTACT (INfrastructure-free TACTical situational awareness)], and the Finnish Geospatial Research Institute (FGI).
In this study the objective is to use the motion context information to create adaptive navigation algorithm. We use machine learning to achieve better results in navigation but do not aim to create the best possible motion context classifier. Improving the navigation result, rather than the most accurate classification, is the main goal of this paper although the classification accuracy and navigation result are definitely related.

We test our algorithm in a realistic tactical scenario with fast paced movement. This test scenario is chosen because we want to show that the used navigation algorithm works in applications where infrastructure-free navigation is necessary: first responder, rescue, and tactical applications. Moreover, in this study we show that context adaptive inertial navigation and context adaptive fusion of visual and inertial navigation is feasible and useful even when the person is moving at a fast pace and carrying a lot of equipment.

In section II we discuss the related scientific work in the fields of motion context recognition, pedestrian inertial navigation and visual navigation. In section III we outline the methods in our context adaptive navigation. In section IV, we discuss the tests we did to evaluate these methods and the test results. In section V we discuss future work and conclude this paper.

II. RELATED WORK

A. Motion context recognition

In general, an activity recognition system consists of a sensing module, a feature processing and selection module, and a classification module [13]. Wearable accelerometers, as well as sensors such as magnetometers, barometers, thermometers and microphones have been used for this purpose [14]–[16]. Many of the sensors used for motion recognition in previous research are included in our infrastructure-free navigation system [1], so it is fairly straightforward to use them as a sensing module in an activity recognition system designed for improving navigation outcome.

The raw sensor outputs are not usually used as such in motion recognition, but are transformed into features using measurements made over some time period having a predefined length. Often these features are time domain features, such as means, variances, medians and other statistical measures of the sensor output [16]. Also different frequency domain features, for example frequency peaks and peak power estimates, spectral centroids and spreads, have been used [13]–[15], [17]. Usually some feature selection needs to be performed, as the best feature set for the problem at hand often can not be known in advance [18].

In general, no machine learning algorithm is better than other. This is known as No Free Lunch Theorem [19]. In practice this means that the best suited algorithm for motion recognition has to be found through testing separately for each problem at hand. In existing research for example decision trees [14]–[18], Bayesian classifiers [15], [16], [18], [20] and Support Vector Machines (SVM) [11], [15], [18], [20] have been used for motion recognition. In this work we use a Random Forest (RF) classifier, that is an ensemble of decision trees. For more details see [19], for example.

B. Zero velocity update

In inertial navigation the simplest method is to integrate the position from acceleration and turn rate observations of an IMU. However, this method suffers from rapid accumulation of error due to the double integration of noisy measurement and lack of absolute position updates.

Mounting the inertial sensor on the foot of the user makes it possible to remove some of the accumulated error [21]. Error reduction is possible because during each step of the user there is a moment when both the ball and heel of the foot are touching the ground and the foot is nearly stationary. The velocity should be zero during the stationary period so a Zero Velocity Update (ZUPT) can be made. This ZUPT can be used as a pseudo measurement in the navigation algorithm and the accumulated error to velocity can be removed.

ZUPT based pedestrian navigation is very dependent on effective detection of the stationary periods. Various methods have been suggested. In [22] several detectors were tested, including: variation of acceleration, magnitude of acceleration, angular rate energy, and the generalized likelihood ratio test, which they derived and named Stance Hypothesis Optimal detection (SHOE). The last two were found the most reliable. Angular rate is used also in for example [23]–[25].

More adaptive ways to detect the stationary period have been explored because a simple threshold does not always work for all movement types such as running and walking. In [26] an adaptive threshold taking into account both velocity and acceleration is tested on running, walking and crawling motions. In [27] a hidden Markov Model is used to segment the parts of human gait.

In [11] the detection threshold is based on the users mode of motion (running or walking), which is first detected from the IMU measurement. In this study we explore the a similar motion context recognition based adaptive ZUPT method but include climbing in the motion contexts and fuse the results with visual navigation.

C. Visual navigation

Due to the limitations of inertial navigation we use a sensor fusion approach and augment the inertial navigation solution with visual navigation. A video camera can be used as a visual gyroscope and odometer [2]. This visual navigation system works through feature extraction to find turn rate and translation from consecutive images. Scale problem caused by using a monocular camera is solved in this method through measuring the height of the camera above ground. This distance remains relatively unchanged if the camera is carried on the user’s chest pointing forwards while the person is walking or running. The visual gyroscope and visual odometer measurements can be used in sensor fusion to support other sensors such as the INS. This type of visual navigation does not need pre-installed infrastructure to operate.
The visual gyroscope and odometer are prone to errors if the field of vision is obscured. Obscuring the field of view may result in the lack of features in images. Lack of features can cause the visual gyroscope and odometer to have erroneous observations or no observations at all. In [28] an error detection method is developed which can detect these instances based on a value called Line Dilution of Precision (LDOP). However, as stated in [12], errors remain after the LDOP analysis and they need to be considered. Moreover, the assumptions made to solve the scale problem do not hold in certain situations, such as when the camera cannot see the ground in front of it.

III. METHODS

In this section we discuss the methods that we use in the tests to obtain motion context recognition and navigation data. First the selection and training of the classifier is described. Then we discuss the details of the navigation algorithm, parts of which were generally described in the section II above.

A. Motion context recognition

In this work we use a Random Forest classifier trained using the test data described in Section IV. We use 374 labeled training samples, containing 330 samples of walking, 9 samples of running and 35 samples of climbing, to train 500 Random Trees used as an ensemble. Each sample consists of features computed from sensor readings obtained during a time period of one second. Before training we replace the missing values in the data using the feature mean in the training data.

B. Adaptive ZUPT threshold

The zero velocity detector used in this work is based on angular rate energy detector similarly as for example in [22]. The test statistic is calculated from the mean of squares of angular velocities in a fixed time window, normalized with the sensor noise level. The length of the time window in this study covered three or four observations from the sensors which were updating at the rate of 100 Hz.

For single threshold comparison we used threshold value $\frac{4}{3} \times 10000$. This value was chosen based on previous experimentation to work adequately for walking. For the motion context adaptive inertial navigation the threshold values were set to $\frac{4}{3} \times 10000$, 20000, and 50000 for walking, climbing, and running, respectively. The walking threshold is the same as in the single threshold test. The running threshold was found based on experimentation with fast paced movement. For climbing we use a threshold value higher than in walking, since the foot may move more unpredictably while climbing compared to walking. Based on trials with the test data we chose not to use a threshold value as high as in running because climbing usually is relatively slow paced movement.

For each second the used ZUPT threshold is determined by the current motion context of the user. The ZUPT threshold will be chosen by the previously trained motion context classifier that categorizes the current context based on the sensor measurements to either walking, running or climbing.
choose to discard the visual data under certain contexts. As mentioned in II-C, the visual data may be unreliable in certain situations, such as when the user is climbing and the camera faces the ladder and the wall. The method presented in this section is illustrated in a flowchart in Fig. 1.

IV. TESTS AND RESULTS

The tests were performed in co-operation with the Finnish Defence Forces. The tests consisted of recording the data of navigation sensors worn by a test person and post processing it. These sensors included three IMUs that were attached on both feet and to the chest of the person. The IMU sensors recorded acceleration and turn rate data for motion context recognition and navigation. In addition to IMUs the person carried forward pointing video camera mounted on the torso of the test person. From this video we extracted visual odometer and visual gyroscope data that is the estimate of translation and turn rate. Test equipment on the test persons is displayed in Fig. 2. In the image the person is also wearing other measurement sensors that were not used in this study.

Our goal is to test how the adaptive navigation algorithms help in a difficult navigation scenario such as first responder, rescue, or tactical operation. The test persons wore tactical equipment to make the movement of the test person more appropriate for the application. The test track started outdoors, immediately entered a building, travelled through several smaller rooms, narrow corridors, one large hall, and ended outside. The track ended on the same location where it started to evaluate loop closure error of the navigation solution. During the route the person also climbed up, moved sideways and then descended on a ladder that covered part of one wall of the large hall. Approximate illustration of the route and the location is shown in Fig. 3. The person travelled the route twice. The first round they walked slowly and stopped occasionally. On the second round they moved faster and sprinted occasionally to resemble a more realistic motion.

One round of the test track was approximately 170 meters in length. Two test persons produced in total three sets of test data each containing two rounds of the test track. The test person 1 did the first and third round of tests. The test person 2 did the second round. The third round was not used for navigation but was used as training data for motion context classification.

A. Motion context recognition

Data was partitioned into windows of 1 s each and inertial and visual features were extracted from these time windows. From the video we determined motion type, the context, that each second of data represented. These features connected to context label, either running, walking or climbing were then used in classification. A Random Forest classifier was trained using the third set of data, done by test person number 1. This model was then tested on the two other datasets, one obtained by test person number 1 and one obtained by test person number 2. Confusion matrices can be seen below on Table I.

<table>
<thead>
<tr>
<th>TABLE I</th>
<th>CONFUSION MATRICES FOR THE RF CLASSIFIER.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Classification for Person 1</td>
</tr>
<tr>
<td></td>
<td>True Class</td>
</tr>
<tr>
<td></td>
<td>Walk</td>
</tr>
<tr>
<td>Walk</td>
<td>372</td>
</tr>
<tr>
<td>Climb</td>
<td>0</td>
</tr>
<tr>
<td>Run</td>
<td>3</td>
</tr>
</tbody>
</table>
Running and climbing are not as well recognized as walking. The trained classifier is used during navigation tests to determine the current motion type to obtain correct ZUPT threshold.

B. Navigation

We tested using the motion context information obtained in previous test with our indoor navigation algorithms. The navigation solution was calculated from both test sets that were used to test the classifier. Each second of data received a context label based on what motion type the classifier finds during navigation. The information of the recognized motion type was used either to modify the threshold value for ZUPT in case of running and climbing motion or to omit the visual navigation data in the case of climbing motion, pictured in Fig. 4.

We processed the navigation data both with the recognized context information and without it to evaluate the difference. With context information we used different ZUPT thresholds for walking, climbing and running motions in the extended Kalman filter for inertial navigation solution. We also omitted camera observations while climbing in the particle filter that fuses the INS solution with visual navigation data. Without context information in the processing, we use only single threshold and rely on the LDOP values to recognize poor visual navigation observations. Each processing in the particle filter produces slightly different result depending on the random number seed used in the processing. The navigation results for the particle filter are computed as the average of 200 processing runs of the particle filter with different random number seeds. Each processing run used 2000 particles.

The results show that the navigation result improved after introducing the context information to the sensor fusion. However, for the first test round, during which the person moves at a slow pace, the results are quite similar with and without the context information. We measured the loop closure error, which is the difference in position with the first and last measurements at the same location. Results are the mean loop closure errors of 200 processing runs. For the slow paced round the mean loop closure error was 5.07 m without context use and 4.87 m with context use for the first test person. The mean loop closure error was 3.89 m without context use and 3.71 m with context use in the slow paced round for second test person. Both had approximately 0.2 m or on average 4 % improvement with context use. Results are compared in the Table II.

There is walking and climbing but no running in this slow paced test. Lack of running explains why the context use does not improve the results more in the slow paced test. The resulting navigation route and loop closure error for single

---

**TABLE II**

<table>
<thead>
<tr>
<th>Movement pace</th>
<th>Test person</th>
<th>No context info</th>
<th>Context info</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slow Person 1</td>
<td>5.07</td>
<td>4.87</td>
<td></td>
</tr>
<tr>
<td>Slow Person 2</td>
<td>3.89</td>
<td>3.71</td>
<td></td>
</tr>
<tr>
<td>Fast Person 1</td>
<td>5.19</td>
<td>4.65</td>
<td></td>
</tr>
<tr>
<td>Fast Person 2</td>
<td>4.72</td>
<td>3.82</td>
<td></td>
</tr>
</tbody>
</table>
processing round is shown as an example in the Fig. 5. This route was obtained by the second test person on the slow paced test round. The plots of the route processed without context use (a) and with context use (b) are quite similar although there is some difference in the loop closure error. Starting direction is arbitrary but same for both plots. This is one result of the 200 processing runs shown as an example of the performance. Route can be compared to actual one taken shown in Fig. 3.

In the second round of testing, where the person moved at a fast pace, the difference between using and not using context information is considerable. For fast paced round the mean loop closure error was 5.19 m without context use and 4.65 m with context use for the first test person. The mean loop closure error in the fast paced round was 4.72 m without context use and 3.82 m with context use for the second test person. The decrease in loop closure error with context use is 0.54 m and 0.90 m respectively or on average 14 %. Results of this fast paced test compared to the results of the previous slow paced test are in the Table II.

One of the 200 processing results from the fast paced test round by test person 2 is shown as an example in Fig. 6. Compared to Fig. 5 the results without context use (a) and with context use (b) are more different. Again the starting direction is arbitrary but same for both plots. Starting location in this figure is different for both plots since it is determined by the end location of the first round. However, this does not affect the result as only the difference between round start and end points is considered.

Most notably in the Fig. 6 the upper right corner of the plot (a) there is a spike in the route. This is due to person running fast during this part of the test. Because of a low threshold value the algorithm does not get ZUPT for a long time causing error to accumulate. When the foot being stationary is finally detected, the navigation algorithm is able to remove some of the accumulated error. However, some of the error remains as can be seen in the loop closure error, the difference between start and end points. The plot for navigation result processed with context use (b) is cleaner and loop closure error is smaller due to the ZUPT being made more often.

V. CONCLUSIONS

We tested our motion context adaptive navigation algorithm in a realistic tactical scenario with two test persons wearing application appropriate equipment. The purpose of the test was to evaluate our navigation algorithm for applications that would benefit most from infrastructure-free navigation: first responder, rescue and tactical applications. The loop closure error results in navigation tests show improvement in the location accuracy when context information is used in the processing. The loop closure error was reduced on average 4 % when the movement was slow and 14 % when movement was fast. While using context makes the system more complex, this complexity is justified. The context information, such as the information whether the user is walking, running, or lying down is already valuable information for example to the command of a rescue operation that can monitor the status of the users. This information should also be used to aid the navigation.

The classification accuracy is good overall but better for walking than for running and climbing. This is likely due to the fact that the training data had many times larger number of walking samples than climbing or running samples. However, the objective of this paper was to create an adaptive navigation algorithm and not to create the best possible motion context classifier. The objective of this paper was achieved because navigation accuracy improved with the adaptive algorithm even though not all running or climbing instances were correctly detected. Every instance does not need to be correctly classified as long as enough of them are recognized so that regular ZUPT observations can be made. Context information for visual navigation seems to be helpful too. Camera is a useful sensor since it is self-contained and can be fused with the INS but has different sources of error. LDOP value does not detect every instance where the visual odometer and gyroscope values should not be used. In some contexts the camera image should not be used at all. For example while the user climbs a ladder or crawls on the ground the field of vision of the camera is too limited. Context recognition
was able to remove some of the poor visual observations not directly detected from the LDOP during climbing. Further research is required to make infrastructure-free navigation system suited for all situations. For example crawling should be added as a context to the system. Crawling motion is problematic for both visual and foot-mounted inertial navigation. The motion needs to be recognized so some other navigation method can be used during crawling. Also using context information for climbing up or down could be used to improve vertical navigation. There are many ways new contexts could be introduced to the current navigation systems. The results of using motion context information in adaptive navigation are promising.

REFERENCES


