Some project information is in Canvas.

You can chose one of the examples there or make up your own.

Either way a written description of your choice ($<<$ 1 page) is required by Dec. 2.

This will allow me to tell you if it is good, no good or not so good.

Groups of 1-3 people only. 2 is best.

Maximum report length is 10 pages, less is ok.

Minimum one reference article in the bibliograpy.

Report and Code are due Jan. 5, 2019.

The projects require you to make several choices up front.

What estimation problem would you like to work on. Many will chose a robotics example such as discussed in the book but other problems might be target or people tracking, signal frequency tracking, motor position/velocity... This also envolves selecting the sensors you will use.

You can simulate the true state and sensor data or you can download data of real sensor data from web sources. In some cases you will be able to collect your own data.

You will select one (or two) estimation method from those discussed in class or some other if you like.

You will implement the estimator in any language (Matlab would be easiest I think).

Then run the data through the estimator and analyse the results.

The report is how I will assess the project. Each student will write a separate report.

Be wise with your effort and document the process so that all your work can be graded.

Some projects are ambitious and end up not working. This is 'ok' as long as you give a good analysis of how you determined it was not possible, what were the limitations and could you measure them... Remember the implementation counts so some documented results even if not up to required performance is needed.

Poorly written reports of brillant work will not get best grade.

Well written reports on good work are a better strategy.

Well written means right organization, good background, clear statements of what was done, insights and explanations are needed.

Fundamental issue in a report is drawing a conclusin.

With no conclusion it gets hard to motivate reading the report. (I will of course anyway:).

Conclusions do not apear by magic at the end but rather come from forming a question at the project planning stage and then designing the project to answer that.

The evaluation is critical to that answer, so form a question that can be answered in an evaluation (normally summarized in a figure).

Question does not have to be new and unknown but something you want to test for yourself.

Some of the Topics from Previous Years:

Visual Tracking of Mario in Super Mario you tube video using the Particle Filter

Predition of motion of a trailer truck from video motion capture with occulded frames using EKF (Combined with Automatic Control Project in EL2421).

Inertial parameter estimation using force torque sensor and a robot arm/gripper with the KF.

Face tracking in video using PF and KF.

Soccer ball tracking in video stream using PF.

Occupancy grid Mapping using a robot equipped with an Ultra sonic sensor (built in the robotics course)

Various extensions of the Labs to EKFSLAM, UKFSLAM, Grading is based on three criteria

1. Amount (Depth/Scope of Experiments) (0-5 points) (this is 0 for literature study)
2. Clarity (The Report) (0-7 points)
3. Insight (bonus added to grade, for example $C- > B$)

The first two critera combine to give a grade according to the chart on next slide. This can get you to B at best. Then the Insight grade, if any, will move you up one or two grade levels. (It pays to show me you learned something doing the project.)

Grading Amount is the first column, Clarity is the first row

	1	2	3	4	5	6	7
0	F	F	F	E	E	D	D
1	F	F	F	E	E	D	C
2	E	E	D	D	C	C	C
3	E	D	C	C	C	C	C
4	E	C	C	C	C	C	B
5	D	C	C	C	C	B	B

1. Amount (0-5 points), This has sub moments:
   - Implementation (0-3 points) A reference value here is implementing EKF SLAM using point features. That is a 1. Implementing several SLAM methods could possibly get to 2 if done well. Implementing some of the harder SLAM methods, using some other features and/or sensors than the lab, possibly real data, and comparing methods could get a 3. Generally creativety gets some reward here.
   - Analysis (0-2 points) This has to do with simulating the right things that show interesting properties of the estimators. For example just using the lab datasets would get you 1 point here. Adding enough interesting own simulations could get you to 2.

2. Clarity (0-7 points), The breakdown to parts is on webpage and the expectations in the ProjectReport-Guide.pdf. Amount will give you no credit if you fail to explain it well.

Here we expect that you read, discuss and cite at least one article (besides course literature and articles on the course webpage). A web reference does not count towards that total of one article but rather a journal or conference paper such as for example an IEEE publication.

The references should be cited with enough description to prove that you have grasped the main point(s). It is much better if they are woven into a story of the State-of-the-art and related directly to the method you will implement.

English grammar however is not part of the grade and these errors will be ignored if the meaning is clear. It is important to have the right content and organization in your paragraphs and sentences.

On the other hand, copy and paste of text (and figure) is always a serious offense. (unless in quotes with citation ref.

3. Insight, That is show that you increased and deepened your understanding. So insight is not paraphrasing a lecture or some other source. Rather more focused on the implementation you did and the experiments you devised.

Why did you do what you did. What did you expect to happen. What did happen. Why? What did you learn about the method (for example, when is it a good or bad choice)?

The connection between measurement uncertainty and estimation uncertainty could be discussed. What other themes from the course can you relate to your example?

The project is suppose to be a first step to doing independent research. So you should learn on your own.

You have access to huge database of good information, Google Scholar from a kth domain.

Generally much better return on time spent on background section than time spent on implementation.

In case you are not aware:

Anything in your report that you did not do yourself must say so and who did it.

Plagiarism is not tolerated. .

This means one can not find a nice bit of text and paste it in. (This is normally real easy to see.)

Each Group must write their own code.

Each student should write up the report separately. So the code and plots will look the same but the text explaining it will not.

Well written technical report? whats that?

Typically one has a flow (possibly section headings) like:

- Intro
- Background
- Method
- Experiments
- Conclusion
- Bibliography

Could also have an appendix with technical information.

- Have both a 'red thread' and a coarse to fine progression.
- What you actually did do (and did not do) should be stated clearly and early (first page) in broad terms. The reader needs to know why for all text that follows (context).
- Often references are just rattled off but much better if they are woven into the story. So presenting an idea an then giving the ref were it came from.

Intro: (Why, What/How)

- Job here is to help the reader focus on what is important in all the rest of the text.
- State clearly what this is all about (Context and frame for the reader).
- Problem statement in conceptual terms (no formulas). Jargon is ok with references but do not go overboard. Better to explain at too simple a level than leave the reader confused.
- By the end the reader knows WHY you are going to say all the stuff that follows.

Background.

- Here you write about the different ways that parts of the problem can be handled including but not limited to the method you chose.
- Can be just a long intro but can be a seperate section.
- Most of the references are typically here but that is not required or always the best way. Some will have to be.
- Usually try not to have formulas or formal algorithms here. Just conceptual.

Method.

- Here you explain in more detail what you did.
- Not the details of the experiment but rather the methodology
- Here you give formulas or algorithm descriptions. (Pseudo code does not count as a description)
- Avoid diary style.

Experiments

- The reader has to understand what you did.
- All choices need to be stated. (normally also motivated)
- Reader should be able to repeat what you did.
- Results that are shown should serve a purpose, ie show that it worked and how well.
- There can be a separate disscussion section or it can be woven in but som analysis of the results is essential.

Conclusions

- What did you learn. (Normally what did the world learn but this is a course project).