

A problem-oriented research plan guide

David Morilla Cabello

January 2026

1 This document

This document provides a brief introduction to my view on proposing research problems. It is by no means a comprehensive study, drawing mainly from my own research experience, discussion with peers, and feedback from my seniors. Many of these ideas are applied intuitively in most labs, but I have not found any written guide explaining the process in detail. Given my background, the insights here are primarily rooted in Robotics, Computer Vision, Control, and AI, with some extension to other areas of Computer Science, so the guidance may not fully apply to other research communities.

When proposing a research project, some labs or researchers use templates with guiding questions. While useful, these brief prompts can be interpreted differently by novice student researchers and often lack examples, advice, caveats, or detailed explanations of what to include. In this document, I aim to share my perspective on proposing research problems thoughtfully and systematically, to help avoid common mistakes.

That being said, I believe research benefits from flexibility and some entropy in the process. While this document provides a framework for defining a research problem, it is not meant to be followed rigidly or as a one-size-fits-all solution. Most of the time, the feedback from supervisors, peers, and mentors should supersede this document. As the project advances, implementations become unfeasible, and results do not match the expectations, the plan will and should vary from the original proposal.

Any feedback or discussion regarding the ideas in this document is greatly appreciated. You can contact me at: `davidmc@unizar.es`

2 Problem

What problem do you want to address? It need not be practical or applied, but it should tackle a current limitation or knowledge gap in the state of the art. This problem forms the foundation of your research proposal, defining success criteria, Key Performance Indicators (KPIs)¹, milestones, potential benchmarks, and relevant literature. If your proposal cannot be clearly linked to a specific problem, it often signals a lack of context or an attempt to propose a method without a well-defined issue. This can result in vague success criteria, weak KPIs, and difficulty situating your work within existing research.

2.1 Formulating your problem

Due to differences in the types of problems addressed by various research communities and publication venues (e.g., conferences and journals), let me distinguish between two broad categories of problems:

- The first category is *application-oriented* problems, where the goal is to achieve a specific outcome, but current tools or methods fall short. These are most common in applied communities such as Computer Vision or Robotics.
- The second category includes *knowledge gap* problems, where the objective is to understand or explain phenomena more deeply. These are more common in fundamental fields such as Behavioral sciences or some AI communities like representation learning.

While every community welcomes both types of contributions, their focus shapes expectations. For example, publishing a Robotics paper usually requires a clear application unless the theoretical insights are particularly strong, whereas demonstrating an application in a theory-driven field rarely counts as

¹KPIs are performance measures used to evaluate success; here, they indicate how effectively the problem is addressed.

a substantial contribution without significant theoretical development. Research can address both types of problems, but the approaches and expected outcomes are fundamentally different.

In my opinion, long-term research should aim at solving both types of questions. Solving a practical problem in depth or applying a theoretical breakthrough to formulate new solutions. This is what seminal papers accomplish. However, many projects assigned to students have limited time, so it is most effective to focus on the common problem types valued by your target community, which sets their expectations.

For application-oriented problems, I find it useful to pose them as affirmative statements that directly address the limitation rather than “research questions”. Conversely, knowledge gap problems can be posed as questions that precisely identify the gap. This distinction is crucial. For instance, “How to map dynamic objects?” isn’t the problem itself; the actual problem is “Mapping dynamic objects,” which usually connects to higher-level problems (e.g., autonomous driving). Similarly, “Interpretability of neural networks” is ambiguous; the problem could be posed as “How can we interpret (a certain behavior of) neural networks?”

Research addressing the first type yields a practical solution, while the second generates additional knowledge that may not have immediate application. Confusing the two can lead to either failing to achieve an application or trying to fill a research gap with only an evidential case. Nonetheless, pursuing either type might yield results relevant to the other. You might find theoretical insights from solving an applied problem or applications for new theoretical insights, but these should not be expected or assumed as a result.

A common trap I have observed is framing problems as “How do I use this new X for Y?” simply because a tool or trend is new. Adopting a novel method is not a problem in itself; methods are tools to solve existing problems. Applying them without a clear objective (problem) rarely leads to productive research. You can propose to use new methods to find solutions, but first identify the clear problem rather than starting with a method and searching for a problem. A compelling illustration is the emergence of photogrammetry methods like Neural Radiance Fields (NeRFs) or Gaussian Splatting. While these innovations inspired numerous efforts, the most productive results came from groups with pre-defined problems. For example, teams focused on Simultaneous Localization and Mapping (SLAM) quickly integrated these methods into their challenges, achieving substantial improvements.

2.2 Narrowing it down

Once you can clearly name your problem, you can consider the ideal solution. What indicates the problem is solved or the knowledge gap filled? At this stage, think informally. Instead of specific experiments or metrics, define success in terms of general performance indicators and milestones. This helps ground the motivation and identify relevant literature to contextualize the problem.

Most problems naturally break into subproblems and vary in **scope and context**, forming a hierarchy. For example, “navigating an autonomous car from A to B” requires solving “planning in dynamic environments” and may include challenges like “planning in real time.” Considering this hierarchy clarifies how your problem fits within existing solutions. Ask yourself: Does your solution improve previous ones quantitatively? Do other solutions address your exact problem, or are you extending existing work in a novel qualitative way?

At this point, three concepts must be clearly distinguished. First is the **problem** that guides your study. However, its definition is strongly influenced by the **related literature**. In some cases, the starting point for defining a problem naturally emerges from prior work in a field of interest, addressing existing limitations. In other cases, you may initially define a problem that has already been addressed, requiring you to refine or reformulate it. Finally, there is the **motivation** of the work, which is reflected in the definition of success metrics and KPIs.

It is important to remember that changes to the problem definition directly affect the success criteria, KPIs, relevant literature, and the core motivation of the research. For this reason, the problem should be clearly defined, grounded in a literature review, accompanied by well-articulated success criteria, and ideally informed by early preliminary results to enable rapid iteration. Conversely, developing a solution without a well-defined problem or context can lead to unproductive efforts, either because the problem has already been solved or because the proposed solution fails to satisfy meaningful success criteria. If the problem is later adjusted to better align with a motivation, related field, or technical method, all previous steps should be revisited to ensure coherence and relevance.

3 Motivation

Time is limited, so you can only focus on so many problems. Solving a problem must be interesting to someone. Whether your proposal is for a supervisor or your future self, motivation explains why the problem matters and keeps the reader engaged. Specify its relevance, importance, and potential implications.

The implications of your work, supported by the problem KPIs and contextualized with related literature, can be quantitative improvements over existing solutions or new qualitative capabilities in broader contexts. Clearly state your research novelty to avoid uninteresting problems or duplicating existing work. A common pitfall here is to frame an implementation of an existing solution as motivation for the project. While implementations with no research novelty may be necessary, they should not be the end goal.

4 Related literature

Finding related literature is a continuous process. Often, it serves as the starting point for identifying a problem, since previous work outlines the state of the art and known limitations. At other times, it is a more exploratory process when approaching a problem for the first time. This process continues through the end of your research, when you validate your results against existing knowledge.

4.1 Research field lifecycle

A crucial part of analyzing related literature is adopting a high-level view of the problem and its proposed solutions. Research fields like visual Simultaneous Localization and Mapping (vSLAM) often emerge when prior developments enable new solutions. For vSLAM, this was the advent of affordable cameras and greater computational power, which allowed early research to apply cameras to the SLAM problem, previously addressed with laser sensors, achieving unprecedented localization accuracy and high-quality 3D reconstructions.

If a research field is promising, it attracts more researchers, bringing both advantages and disadvantages. The problem becomes well-known, requiring less motivation and benefiting from standard success criteria, metrics, and tools. However, competition for novelty intensifies, often leading to saturation where publishing depends on resources and speed. Consider these dynamics when selecting a research problem. Ultimately, the cycle produces papers that establish the problem under a set of assumptions as solved, with solutions or knowledge becoming standards and shifting interest to higher-level problems or more challenging assumptions (exploring qualitative variations).

4.2 Why is it not done?

“Because none tried” might be the easy answer. However, taking some time to try to answer more specifically might save you from infeasible, very difficult, or tricky problems. It may be that limitations stated by the previous literature are either more difficult or less interesting than they seem. Otherwise, they would have been included in the published work.

4.3 Measuring scope

When evaluating related literature, clearly define the scope of existing solutions for the problems they address. Papers typically highlight their contributions and performance within this scope, but rarely solve the problem in its entirety, often explicitly stating assumptions or limitations. Ask: How do current solutions tackle the problem? Which specific aspects are addressed? What claims are made, and do the reported results—given the chosen metrics and assumptions—truly solve the problem, or only a simplified version of it?

If the problem’s success indicators are well-designed, you can assess how far current methods are from the optimal solution. Think in terms of what Nvidia calls the “Speed of Light” performance, which represents the top achievable measure. For example, in image classification, the ideal solution would perfectly classify a representative set of images.² If results are already sufficient under established assumptions,

²This ideal is generally unattainable due to limitations. In this case, computing, data constraints, or inherent task ambiguity.

does it make sense to aim for quantitative improvements? Are there qualitative improvements to propose? This may keep the general problem but changes the subproblem, requiring a revision of metrics, milestones, and motivation.

5 Solution proposal

Only now are you ready to propose a solution. By this stage, you have clearly defined your problem and its success criteria. You understand how existing solutions fail to address your specific problem formulation, provide limited or relaxed solutions, or could potentially be extended to a broader or different scope.

If possible, discuss your problem with your peers and your supervisor. They can provide valuable perspectives, highlight overlooked aspects, or help refine your motivation. Their experience may also guide you to relevant literature from adjacent or distant communities addressing variants of the same problem.

Proposing a solution marks the start of the inventive process and the creation of new knowledge. Several strategies, however, can greatly improve the efficiency and focus of your research.

5.1 What tools do you already have?

When formulating your solution, assess the tools and resources at your disposal. Ask: Are there public or self-developed algorithmic implementations? Are suitable datasets available if using a learning method? Is there a framework for simulated experiments? Does your lab have the necessary setup (e.g., robots, hardware)? Try to find *low-hanging fruits* and consider major system-building efforts only if they provide significant value; in such cases, the construction process itself may become a problem to tackle, and you could pivot your proposal accordingly.

5.2 Proposed solution

Yes, the order is correct: first assess existing resources for low-effort solutions, then develop your remaining wish list. Leveraging available resources before building an entire framework from scratch is crucial. With this foundation, you can now propose a solution. Previous sections address “What” you want to work on; the method addresses “How”. The approach depends on the problem, but a formal thought process typically follows these steps:

- **Informal Solution Proposal:** Begin by intuitively considering solutions that seem logical. This is the brainstorming phase where you explore initial ideas. A word of caution, though: even if an idea is logical, do not bet on it before you actually test it.
- **Formal Proposal:** Define specific algorithms, theorems, or tools (e.g., deep learning models) to address the problem.
- **Your Thesis:** The central claim of your work. For practical problems, it asserts that your method effectively solves a defined version of the problem; for theoretical work, it provides your primary answer to a knowledge gap or open question.
- **Your Hypothesis:** A specific, testable prediction derived from your thesis. It proposes a measurable effect and guides experimentation. One thesis may lead to multiple hypotheses, each targeting a different aspect of the claim.

Even in practical cases, adhere to your problem’s definition and frame your claims in terms of a Thesis and Hypothesis. This structure naturally outlines your core arguments and provides a basis for experimental validation, creating a cohesive narrative. Examples for different scenarios:

Practical scenario: Semantic segmentation. Start from an intuitive insight, such as: *“Attention-based neural network architectures have successfully modeled spatial relationships in language. Perhaps they are useful for image-based processing.”* From this, a research thesis might be: *“Attention-based neural network architectures improve semantic segmentation performance.”* A derived hypothesis could then be: *“Our novel Vision Transformer (ViT) architecture will achieve higher mean Intersection over Union (mIoU) than state-of-the-art CNNs when evaluated on the Cityscapes dataset.”*

Theoretical scenario: Why do ViT perform better? To understand the reason behind the performance gain, a thesis could be: *“The superior performance of Transformer-based architectures in*

semantic segmentation arises from their enhanced ability to capture long-range contextual dependencies compared to CNNs.” The associated hypothesis could be: *“Transformers will demonstrate a higher degree of long-range feature interaction, as measured by [metric, e.g., attention span analysis or feature correlation across distant pixels], leading to improved contextual understanding in semantic segmentation tasks.”*

5.3 What am I going to do?

Although experimental design is outside this document’s scope, a clear problem definition makes it far easier to specify your proposal and the experiments needed to validate it. What will you actually implement? What experiments will you run? Describe these ideas at a high level, since it’s normal not to know the final experiments a priori. But doing so demonstrates the proposal’s experimental feasibility. Also propose an incremental plan: begin with simple, toy examples to understand both the problem and how your proposed solution might fit, and progress incrementally to the final, ideal evaluation. These initial ideas should foster discussion with your supervisor on why they are necessary (or could be assumed as already validated) and relevant for the solution.

5.4 What tools do I need?

Once the proposed method is defined, the next step is to identify the tools, algorithms, and experimental components required for its implementation. This process establishes a clear inventory of what must be built or acquired to run the experiments and validate the claims. Before finalizing this list, it is important to reassess the available resources. Modest adjustments to the proposal may allow existing components to be reused or extended, ensuring that effort is concentrated on aspects that are genuinely novel and scientifically meaningful.

5.5 Expected Results/Outcome

Experiments should be designed to directly test the stated claims. To ensure this, it is good practice to define the ideal outcome before conducting the experiment. **Caution:** “Expected” does not mean tailoring the experiment to confirm a claim, as this would constitute ill-posed research. Rather, the key question is whether the anticipated outcome would genuinely support the claim. If the answer is negative, the experimental design is flawed and should be revised early, rather than adjusting interpretations after the fact. In general, it is preferable to set modest, well-justified expectations and be positively surprised by the results, rather than the opposite.

5.6 Milestones and indicators

Clear milestones and evaluation indicators are essential for validating progress, particularly during uncertain phases of a research project. When direction becomes unclear, returning to these predefined goals helps avoid detours toward efforts that are misaligned with the core research problem.

In software development, a Functional Specification Document (FSD) defines a program’s requirements independently of any specific implementation. It serves as the authoritative reference for development and establishes the functional tests used to validate the final system. While developers may propose changes, the specification itself is rarely altered without strong justification.

Analogously, the milestones and indicators of a research project should remain independent of the chosen implementation. Modifying experiments or success criteria in response to unsatisfactory results **compromises the original problem formulation**. These benchmarks are designed to capture the essence of the problem and must remain the objective measures of progress. This risk is exemplified by the *curse of simulation*, where the same researcher designs both the simulation environment and the method under evaluation. When both evolve together, the simulation may inadvertently favor the method, masking its limitations and introducing bias.

Maintaining a stable problem definition, fixed milestones, and consistent success criteria is therefore critical. These anchors preserve alignment with the research objectives and ensure that results remain valid, unbiased, and representative.

5.7 What if it goes wrong

During your problem proposal, try to identify cases where implementation may be unfeasible or where results might turn out poorly. Create milestones that expose these risks as early as possible, and consider brief contingency plans that offer a clear pivot if needed. Addressing potential failure points upfront can save you valuable time, especially as deadlines approach.

6 Acknowledgements

This guide was inspired by my experiences in several research labs. In particular, the labs of Prof. Marija Popović and Prof. Cyrill Stachniss introduced me to a structured and methodological approach to developing research ideas. I am also grateful to the Vision for Robotics Lab, where Margarita Chli and Luca Teixeira helped me define my first research project. Finally, I would like to thank all my co-authors, mentors, with emphasis on Eduardo Montijano, my PhD supervisor, and peers who contributed to this document, either indirectly by sharing their research processes or directly by providing valuable feedback during its writing.

Research Plan

Topic: Cool research topic

For: David Morilla Cabello

Date: 04.06.2023

Which problem do I want to address or solve?

**Is it a practical solution or filling a knowledge gap?*

Why is this relevant? Why should I read this? Why should I care?

**What are the potential implications of solving this problem? Quantitative improvements? New capabilities? Broader impact?*

4.2 Proposed solution described in a general way

**What are the claims of the proposed solution?*

What am I going to do?

**List required software implementations, theoretical work (research, proofs...), or possible experiments that come to your mind. No need for a thorough plan, keep it at the highest level, and we will work from it. LIST SPECIFICALLY WHAT ARE MISSING/CRITICAL PARTS!*

Expected Results / Outcomes

What are the expected outcomes of your experiments? Do these outcomes validate your claims? What is the minimal success criterion to consider your work convincing?

Measures of Success, Milestones, and Good Indicators

List your major milestones (e.g., problem formulation, prototype, experiment 1, evaluation, etc.) What objective indicators or metrics will you use to measure progress at each milestone? Are your milestones aligned with your original problem definition and success criteria?
