LaTeX BEAMER Theme

a simple and clean slide template

Presenter Name

Complete Institute Name

October 16, 2025

Table of Contents

Table of Contents

Metadata Settings

and beamer class settings

BEAMER class: The class is default to `presentation` mode. For printing handouts, use `\documentclass[handout]{beamer}` instead. Handout mode Incomplete yet.

Metadata fields: Several metadata fields can be set in the preamble:

- `\title[short title]{full title}`
- `\subtitle{subtitle}`
- `\author[short name]{full name}`
- `\institute[short name]{full name}`
- `\date{date}`
- `\logo{graphic}`
- `\titlegraphic{graphic}` (optional)

Frame Settings

Frame options: A `frame` environment can take several options:

- `plain`: removes header and footer.
- `noframenumbering`: excludes the frame from slide numbering.
- `label=label_name`: assigns a label to the frame for hyperlinking.
- `t`, `c`, `b`: aligns the content at the top, center, or bottom of the frame.

No frame number indexing: To remove frame number indexing at both the navigation bar and the footer, wrap the `frame` environment with the `noframenumber` environment defined in the theme.

Frame title and subtitle: Use the `\frametitle{title}` and `\framesubtitle{subtitle}` commands within a `frame` environment to set the title and subtitle of the frame, respectively.

Title Page & TOC Slides

Title Page: The title page is created using the `\titlepage` command within a `frame` environment. It is recommended to use the `plain` and `noframenumbering` options for the title page frame.

Table of Contents Slide: A table of contents (TOC) slide is created using the `\tableofcontents` command within a `frame` environment. It is also recommended to use the `noframenumbering` option for the TOC slide. Note that a TOC slide highlighting the current section is automatically added at the beginning of each section.

Objectives

Scope

Sample Block Title

This block presents a key concept that is crucial for understanding the topic.

Sample Alert Block Title

This block presents a more alarming key concept that is crucial for understanding the topic.

Actors & Features

Actors:

Features:

Contributions

Scientific Contribution

Real-world Contribution

Table of Contents

Advancements

Research gaps

Research gap

⇒ **Concluding statement.**

Table of Contents

Overview

Figure. The caption of the figure.

Sample Process

Algorithm Pseudocode

- **Goal:**
- **Result:**
- **Step:**
- **Scope:**

Table of Contents

Prototyping

GitHub repository:
Demo Website:

Figure. The caption of the figure.

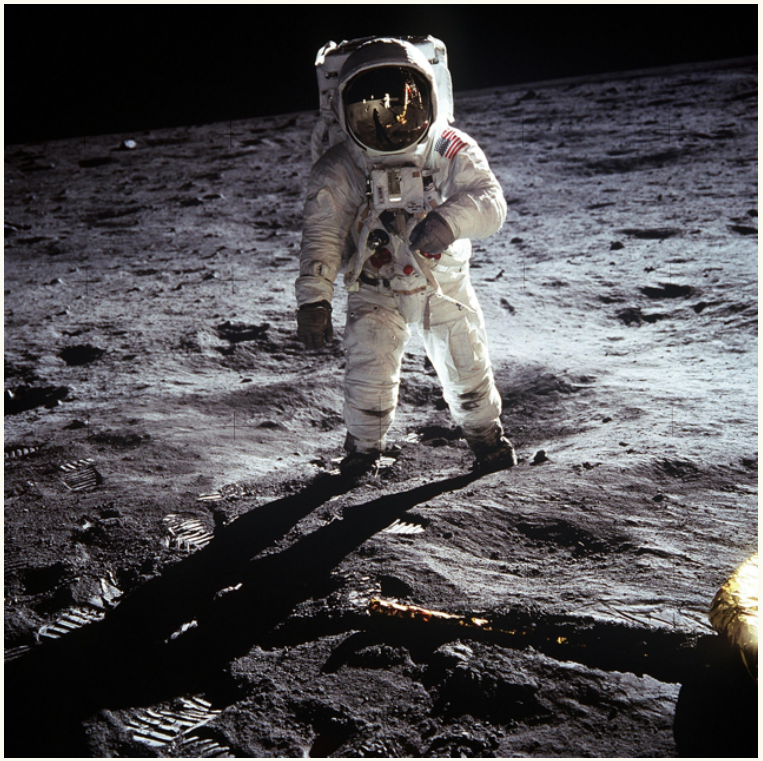

Figure. The caption of the figure.

Table of Contents

Limitations

⇒ **Concluding statement.**

Comparison

Table. Comparison of different methods (✓: YES, ✗: NO).

	Your Method	Method B	Method C	Method D	Method E	Method F
Feature 1	✓	✓	✗	✓	✗	✓
Feature 2	✓	✗	✓	✓	✓	✗
Feature 3	✗	✓	✓	✗	✗	✓
Feature 4	✓	✓	✗	✗	✓	✗
Feature 5	✗	✗	✓	✓	✗	✓
Feature 6	✓	✗	✓	✗	✗	✗

Table of Contents

Demonstration

Process A

Scenario 1

Scenario 2

Process B


thank
you!

Scope

Back to Objectives

Formalizing - `Sample` Algorithm

Back to `Sample` process

Algorithm (Result) ← `Sample`(Input1)

Require: Input1 is a predefined parameter.

```
1:  Set ← ∅
2:  for element ∈ Input1 do
3:      if Condition(element) is true then
4:          Set ← Set ∪ {Process(element)}
5:      else
6:          continue
7:      end if
8:  end for
9:  Intermediate ← Transform(Set)
10: return Result
```

Formalizing - `Sample` Pseudocode

Back to `Sample` process

Algorithm (Result) ← `Sample`(Input1)

Require: Input1 is a predefined parameter.

```
1:  Set ← ∅
2:  for element ∈ Input1 do
3:      if Condition(element) is true then
4:          Set ← Set ∪ {Process(element)}
5:      else
6:          continue
7:      end if
8:  end for
9:  Intermediate ← Transform(Set)
10: return Result
```

References I