

Análisis del Sistema de Inferencia Difuso para Detección de Bordes en Imágenes

Kevin Esguerra Cardona

29 de agosto de 2025

Resumen

Se analiza el FIS utilizado en el pipeline de detección de bordes para resolver sudokus. El informe describe las variables lingüísticas, las funciones de pertenencia, la base de reglas y la estrategia de desfuzzificación. Se justifican los parámetros y se presentan figuras de las funciones y la superficie de salida.

1. Introducción

El pipeline inicia con un FIS que fusiona los gradientes de la imagen para generar un mapa de bordes suave y robusto a ruido.

2. Diseño del FIS

2.1. Variables lingüísticas

- **I_x, I_y**: gradientes en los ejes x e y , con universo $[-1, 1]$.
- **I_{out}**: intensidad de borde, con universo $[0, 1]$.

Las funciones “zero” y “nonzero” son gaussianas con desviaciones $\sigma_x = \mathbf{s}x$ y $\sigma_y = \mathbf{s}y$.

2.2. Funciones de pertenencia

- **Entradas**: $\mu_{\text{zero}}(v) = e^{-(v/\sigma)^2/2}$, $\mu_{\text{nonzero}}(v) = 1 - \mu_{\text{zero}}(v)$.
- **Salida**: Triángulos $\text{trimf}(x; 0, 0, 1)$ para “white” y $\text{trimf}(x; 0, 1, 1)$ para “black”.

2.3. Base de reglas

Regla	Antecedente	Consecuente
1	$I_x = \text{zero} \text{ AND } I_y = \text{zero}$	$I_{out} = \text{white}$
2	$I_x = \text{nonzero} \text{ OR } I_y = \text{nonzero}$	$I_{out} = \text{black}$

El operador AND usa mín y el OR usa máx.

3. Elección de parámetros y límites

Los valores σ_x y σ_y (por defecto 0.1) determinan el ancho de las gaussianas, controlando sensibilidad a bordes débiles. Limitar el dominio de entrada a $[-1, 1]$ normaliza los gradientes, mientras que $[0, 1]$ para la salida permite representar bordes blancos (0) o negros (1) con transición suave.

4. Desfuzzificación y valores extremos

La salida se desfuzzifica mediante centroides. Para acelerar, se emplea una LUT precomputada con 256 niveles. El valor mínimo (0) representa ausencia total de borde, y el máximo (1) corresponde a detección intensa de borde.

5. Figuras

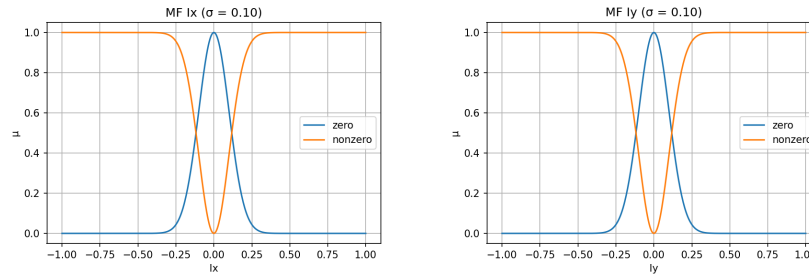


Figura 1: Funciones de pertenencia para I_x e I_y .

6. Conclusiones

El FIS diseñado es compacto pero efectivo: dos reglas y funciones simples permiten detectar bordes robustos. La parametrización con σ_x, σ_y ofrece control fino y la LUT garantiza eficiencia en la evaluación.

Referencias

- [1] Archivo `tinyfuzzy.py`, implementación del FIS.

