

Método para detección y seguimiento de objetos con aplicaciones en Realidad Aumentada

Christian Nicolás Pfarher

Director: Enrique Marcelo Albornoz

Co-Director: Cesar Martínez

**Proyecto Final de Carrera
Ingeniería en Informática**

*Facultad de Ingeniería y Ciencias Hídricas
Universidad Nacional del Litoral*



sinc(i)

Contenido

- 1 Introducción
- 2 Método propuesto
 - Detección de movimiento
 - Extracción de Características
 - Correspondencia de puntos entre imágenes
 - Transformación y RA
- 3 Experimentos y Resultados
 - Entorno de pruebas e imágenes
 - Experimentos
 - Implementación de prototipos
- 4 Conclusiones y trabajos futuros

Contenido

1 Introducción

2 Método propuesto

- Detección de movimiento
- Extracción de Características
- Correspondencia de puntos entre imágenes
- Transformación y RA

3 Experimentos y Resultados

- Entorno de pruebas e imágenes
- Experimentos
- Implementación de prototipos

4 Conclusiones y trabajos futuros

Realidad aumentada

Un sistema de realidad aumentada (RA) reemplaza parte del mundo real con objetos virtuales, los cuales parecen coexistir en el mismo espacio que el ambiente real.

- Introduce elementos virtuales en una escena captada de un entorno real.
- Trabaja interactivamente y en tiempo real.
- Detecta y realiza un “seguimiento” de objetos reales y virtuales entre sí.
- Realidad Aumentada no es Realidad Virtual.
- Según el grado de realismo o artificialidad:



Sistemas y métodos para detección en RA

Tipos de sistemas

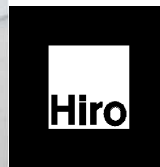
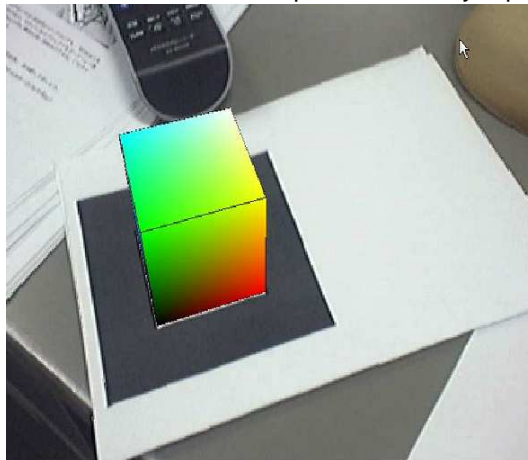
- Basada en marcadores.
- **No basada en marcadores** → características naturales de los objetos.

Métodos para detección y seguimiento de objetos

- Seguimiento basado en localización (GPS, acelerómetros, giroscopios, etc.)
- **Seguimiento óptico** → análisis e identificación de características a partir de la imagen.
- Una combinación de los dos anteriores.

Aplicaciones - Ejemplos

Aplicaciones - Ejemplos





Motivación

- Desarrollo de un software propio de RA adaptable a aplicaciones específicas (comerciales, educativas, lúdicas, etc.)¹
- No todas las aplicaciones trabajan en tiempo real.
- Muchos utilizan marcadores artificiales.
- Distribuidos bajo licencias privativas o costosos.
- Se trabaja con una tecnología que se encuentra en auge en estos tiempos.
- El reconocimiento de objetos puede ser aplicado a diversidad de temáticas.

¹SINC: Centro de Investigación en señales, sistemas e inteligencia computacional

Objetivos

- Diseñar y desarrollar un método reconocedor y seguidor de objetos planos en el flujo de video tomado por una cámara web estándar, sobre un ambiente controlado.
- Implementar el método en un algoritmo computacional que sea multiplataforma.
- Optimizar el procesamiento para aplicarlo en tiempo real.
- Implementar una aplicación prototipo específica (en el área de turismo, educación, publicidad, juegos u otros).

Contenido

- 1 Introducción
- 2 Método propuesto
 - Detección de movimiento
 - Extracción de Características
 - Correspondencia de puntos entre imágenes
 - Transformación y RA
- 3 Experimentos y Resultados
 - Entorno de pruebas e imágenes
 - Experimentos
 - Implementación de prototipos
- 4 Conclusiones y trabajos futuros

Detección de objetos

- Buscar en una imagen o un video (secuencia de imágenes) un objeto particular dado.
- Seleccionar algunos puntos como características distintivas del objeto en la imagen.
- La búsqueda de estos puntos, se puede realizar mediante detectores de puntos claves.
- Se representa la vecindad de cada punto de interés detectado mediante un vector descriptor.
- Se buscan las correspondencias entre los vectores descriptores de las imágenes.

Etapas

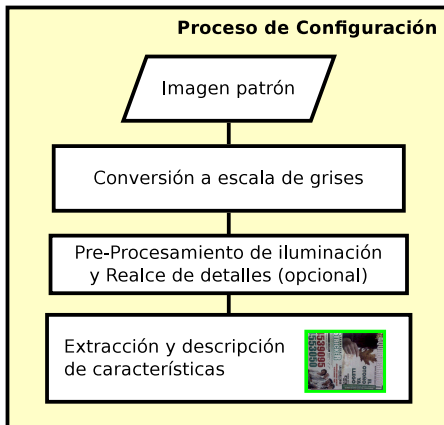
Configuración

- Se realiza una sola vez al inicio del algoritmo.
- Se registra la imagen que posteriormente se detectará en el flujo de video.

Ejecución

- Captura de un frame del flujo de video proporcionado por la cámara web.
- Procesamiento para detectar el objeto registrado en la Configuración.
- Superposición de un objeto virtual para enriquecer la realidad.

Etapa de configuración





Conversión a escala de grises

Formula perceptualmente ponderada:

$$I = 0,299R + 0,587G + 0,114B.$$

Pre-procesamiento de realce de detalles y mejora en iluminación

- Transformación logarítmica $s = \log(1 + r)$.
- Ecualización del histograma.
- Filtrado pasa altos:

$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^a w(s, t) f(x + s, y + t) \quad w = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

- Filtrado de alta potencia:

$$g(x, y) = (A - 1)f(x, y) + PA(f(x, y)) \quad \text{con } A = 2; \quad \text{kernel} = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

- Ecualización del histograma y posteriormente filtrado de alta potencia.

Detección de la región de interés

Determinación de una zona de interés para realizar la extracción de características.

- Diferencia de imágenes (F : cuadro del flujo de video).

$$D(x, y) = |F_t(x, y) - F_{t-1}(x, y)| \quad \forall x, y \in F.$$

- Umbral Binario.
- Erosión $\times 2 \rightarrow$ eliminar puntos aislados.
- Dilatación $\times 2 \rightarrow$ recuperar eliminación de objetos de interés.
- Rectángulo delimitador mínimo (BR).
- Umbral sobre el área de BR

Detección de la región de interés



Frame Previo

Frame Actual

Diferencia absoluta

Umbral binario

Erosión $\times 2$ Dilatación $\times 2$

BR

Resultado de los procesos aplicados para la detección de movimiento.

Detección de puntos claves y descriptor

SURF (Detector rápido de características robustas)

- Detector y descriptor de puntos claves.
- Basado en **SIFT** (Transformación de características invariante a la escala).
- Gran velocidad de cálculo (más rápido que SIFT) con tolerable pérdida de robustez.

El problema de cambio de escala en correspondencias de puntos

- Los objetos fotografiados a diferentes distancias, aparecen de diferente tamaños.
- Buscar correspondencias utilizando un número fijo de píxeles vecinos → las intensidades no coincidirán.
- Definir un área vecina al punto clave con la misma información visual.

SURF

- Invariante a escala: asigna un factor de escala a cada punto clave.
- Invariante a rotación: asigna una orientación a cada punto clave.
- Determinante de la matriz hessiana para la determinación de la localización y escala de los puntos → umbral
- Vector descriptor N dimensional (64 elementos) para cada punto clave.
- Cuanto más similares sean 2 puntos característicos más cercanos serán sus descriptores.

Búsqueda de correspondencias

Grado de similitud

Grado de similitud entre vectores mediante distancia euclídea.

- NNS (Búsqueda del vecino más cercano).
- K -NN (K vecinos más cercanos).
- Búsqueda aproximada mediante KD -tree o multiple KD -tree aleatorio.

Búsqueda de correspondencias

- Cálculo de la distancia euclídea entre los vectores característicos.
- Aplicación de un umbral para determinar las potenciales correspondencias válidas.
- Reducción de correspondencias espurias.

Transformación proyectiva - Homografía

- Relación proyectiva entre dos imágenes de una misma escena tomadas desde diferentes puntos de vista.
- La relación entre las dos imágenes es una homografía si se cumple la ecuación $\mathbf{x}' = H\mathbf{x}$:

$$\begin{bmatrix} x'_i \\ y'_i \\ z'_i \end{bmatrix} = H \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \implies \begin{bmatrix} x'_i \\ y'_i \\ z'_i \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix}.$$

- Los puntos en una vista pueden ser convertidos a la segunda vista mediante H .

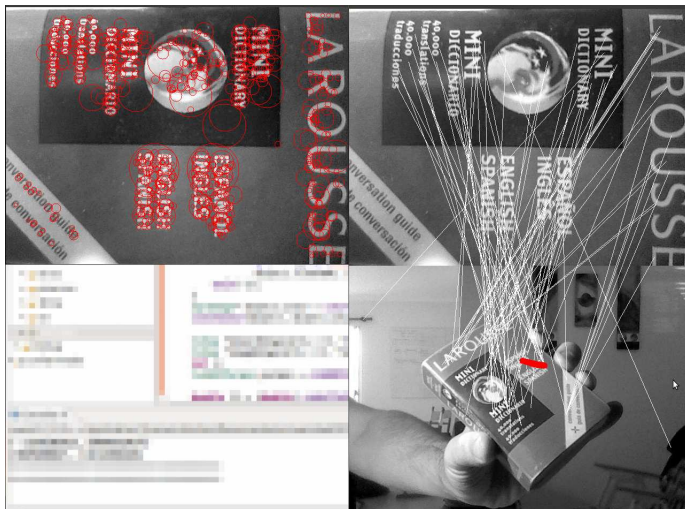
Estimación de H

- Se pueden tener pares de correspondencias que no son válidos o tener más de 4 pares de correspondencias necesarios para calcular H .
- Se procede mediante una estimación de la homografía.
- RANSAC (*random sample consensus*) puede hallar la mejor solución aproximada minimizando el error y depurando coincidencias válidas y espurias iterativamente.

Homografía - Detección



Homografía - Detección fallida



Validación de transformaciones con H

Rechazar transformación fallida

- Comprobar convexidad.
- Distancia entre vértices:

$$(|\alpha_x - \gamma_x| < \Delta X) \vee (|\beta_x - \lambda_x| < \Delta X) \vee (|\alpha_y - \gamma_y| < \Delta Y) \vee (|\beta_y - \lambda_y| < \Delta Y)$$



Polígono cóncavo

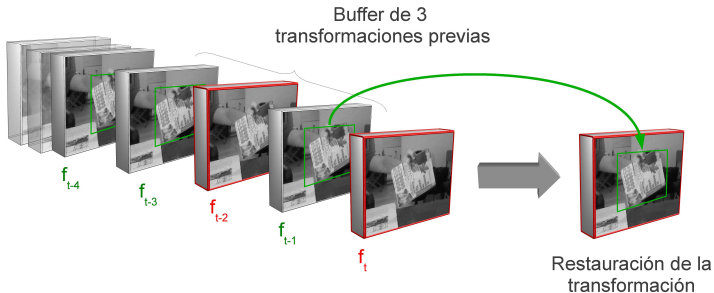


Polígono convexo

Transformaciones fallidas con la matriz H .

Condición de presencia previa

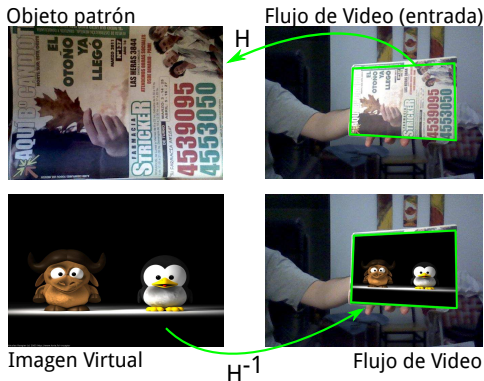
- Restauración de la última transformación válida sobre las últimas tres imágenes procesadas.
- Ante la no detección de una transformación durante tres frames consecutivos, se deja de superponer el objeto.



Esquema de restauración de la última transformación válida.

RA en el flujo de video

- Se utiliza H^{-1} para superponer la imagen virtual en la perspectiva correcta en el flujo de video.
- Existe una interpolación debido a la transformación de píxeles.



Esquema de transformación perspectiva utilizando la homografía H .

Contenido

- 1 Introducción
- 2 Método propuesto
 - Detección de movimiento
 - Extracción de Características
 - Correspondencia de puntos entre imágenes
 - Transformación y RA
- 3 Experimentos y Resultados
 - Entorno de pruebas e imágenes
 - Experimentos
 - Implementación de prototipos
- 4 Conclusiones y trabajos futuros

Definición de imágenes

- **Imagen patrón:** detectar y seguir en cada fotograma del flujo de video. En su lugar se superpone el objeto de realidad aumentada. Restricciones prácticas:
 - 1 tamaño de 640×480 píxeles,
 - 2 condiciones de iluminación adecuadas,
 - 3 imagen debe ser rica en detalles,
 - 4 plano de la imagen perpendicular al lente de la cámara al momento de la captura.
- **Imagen objetivo:** es un fotograma del flujo de video, adquirido con la cámara web en tiempo real.
- **Objeto de realidad aumentada:** imagen, video, texto, etc.

Herramientas

Software

- OpenCV 2.3.1.
- C/C++.

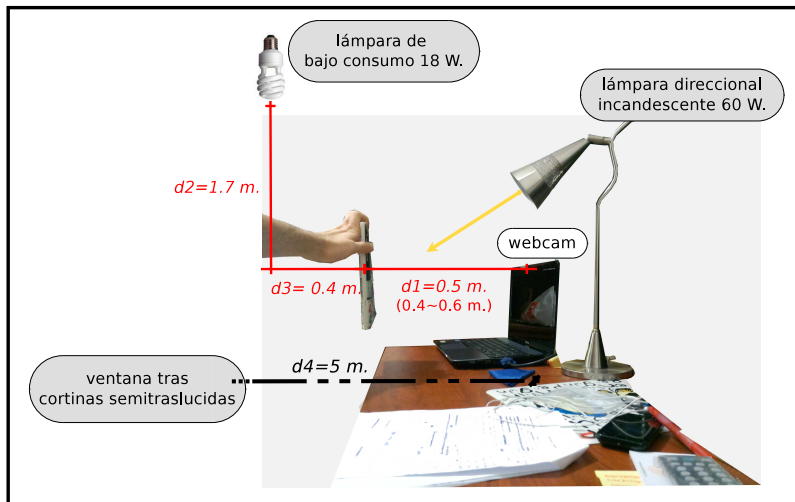
Hardware

- Webcam: Resolución de 640×480 píxeles.
- Procesador Intel® Core 2 Duo 2.2 Ghz., 4Gb. RAM.

Imagen patrón y condiciones de iluminación

- Imagen patrón: tapa de una revista de 22×17 cm.
- Tres condiciones de iluminación diferentes:
 - 1 **Iluminación normal** (B_N):
 - lámpara de bajo consumo de 18 W (90 W. de una lámpara incandescente).
 - 2 **Iluminación alta** (B_H):
 - (B_N) + iluminación direccional con lámpara incandescente de 60 W.
 - 3 **Iluminación baja** (B_L):
 - Dificultad para la lectura de un documento.
 - Lámparas apagadas.

Entorno ambiental



Esquema del ambiente en el que se realizaron las pruebas.

Experimentos

Experimento 1: costo computacional y detección de puntos

Evaluación del costo computacional y detección de puntos claves bajo las condiciones de iluminación B_N , B_H y B_L .

Experimento 1

- Se capturó la imagen patrón en las tres condiciones de iluminación.



Imagen patrón con condición B_N .

Imagen patrón con condición B_H .

Imagen patrón con condición B_L .

Imágenes obtenidas para condiciones de iluminación diferentes.

- Se aplicó a cada una de las imágenes operaciones para realce de detalles y mejora en la iluminación.

Condición B_L

Imagen patrón con Ilum. B_L 

Transformación logarítmica



Ecuilización



Filtrado pasa altos



Filtrado de alta potencia



Ecuilización + filtrado de alta potencia

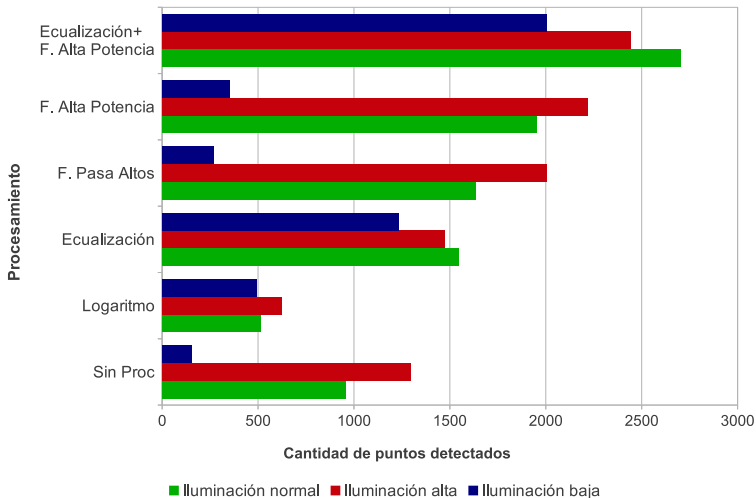
Tiempos y cantidad de características

Umbral hessiano: 700 e igual para las tres condiciones de iluminación.

Tiempo de procesamiento en milisegundos y cantidad de características sobre la imagen patrón en condiciones de iluminación B_N , B_H y B_L .

	t [ms]	B_N	B_H	B_L
Sin Proc	0,00	958	1297	154
Logaritmo	7,15	702	950	83
Ecualización	0,70	1546	1472	1233
F. Pasa Altos	1,26	1633	2006	269
F. Alta Pot.	3,10	1952	2216	353
Ec.+F. Alta Pot.	4,31	2704	2443	2002

Cantidad de características



Experimento 1: conclusiones

- En condiciones B_N y B_H resultados aceptables sin pre-procesamiento de iluminación o realce de detalles.
- Aplicar técnicas repercute negativamente en el tiempo de procesamiento pero aumenta la cantidad de puntos detectados.
- Determinar técnicas a utilizar y umbral hessiano:
 - ¿Qué se desea priorizar: calidad en la detección o velocidad de ejecución?
 - ¿Cuáles son las condiciones del ambiente?

Experimento 1: conclusiones

Prueba en condición de iluminación baja (B_L).

- En la práctica se presentan casos de detección “intermitente” o nula en condición B_L .
- La ecualización o ecualización+filtro de alta potencia mejoran la detección.
- Aumentar la cantidad de características produce una mejora en la detección.
- Una variación del umbral no produce resultados significativamente mejores.

Experimento 2

Experimento 2: costo computacional del método propuesto

Evaluación detallada del costo computacional en etapa de ejecución, considerando los pasos del algoritmo propuesto.

- Tiempos que insumen los procesos del método.
- 2 pruebas en condiciones de iluminación B_N y B_H .
- Sin técnicas de “Pre-Procesamiento de iluminación y realce de detalles”.

Pruebas

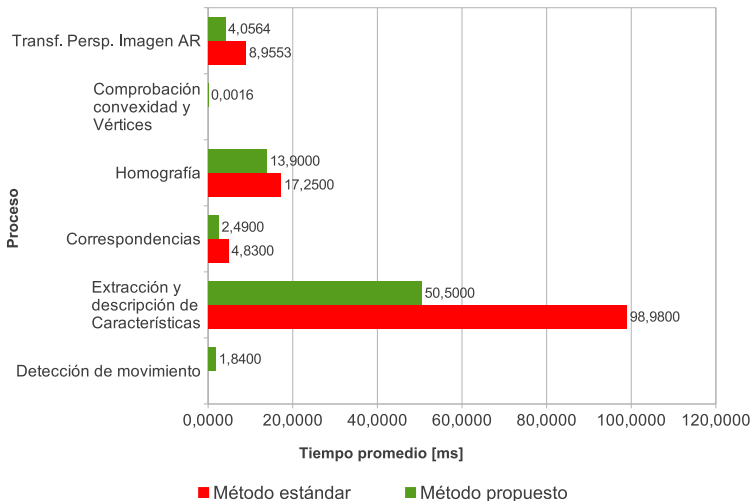
Prueba 1 (P1)

- Duración: 1:50 minutos.
- Umbral hessiano: 3500.
- Imagen con condición B_N .

Prueba 2 (P2)

- Duración: 1:35 minutos.
- Umbral hessiano: 5000.
- Imagen con condición B_H .

Tiempos de operación de procesos en P1



Conclusiones P1

Objetivas

Tiempos:

- Método estándar: 6,93 FPS promedio.
- Método propuesto: 28,35 FPS promedio.

Subjetivas

- Eliminación de transformaciones fallidas.
- Reducción de detecciones intermitentes.
- Mayor fluidez en la reproducción.

Imágenes prueba 1

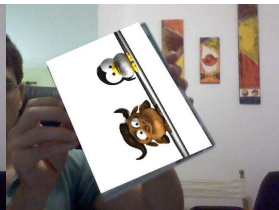
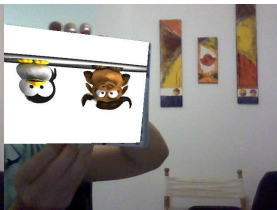
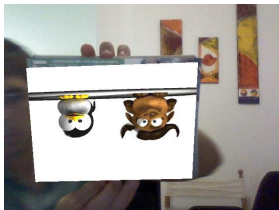
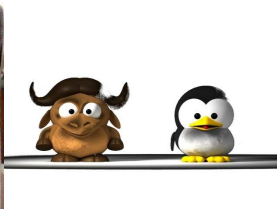


Imagen patrón, objeto de RA y secuencia de imágenes (no sucesivas) con enriquecimiento de la realidad.

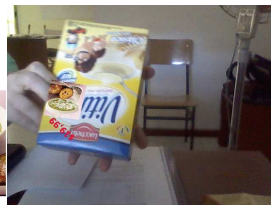
Prototipos

- Se enriquece la realidad mediante una imagen sobre la tapa de una revista.
- Proveer información inherente a un producto comestible de forma publicitaria: video en condición B_N .

Prototipo publicitario



\$19,99



Contenido

- 1 Introducción
- 2 Método propuesto
 - Detección de movimiento
 - Extracción de Características
 - Correspondencia de puntos entre imágenes
 - Transformación y RA
- 3 Experimentos y Resultados
 - Entorno de pruebas e imágenes
 - Experimentos
 - Implementación de prototipos
- 4 Conclusiones y trabajos futuros

Conclusiones

- Se ha desarrollado un método para detectar objetos planos (sin marcadores) en un ambiente natural sobre imágenes obtenidas con una cámara web estándar.
- Se han propuesto técnicas simples para realce de detalles e iluminación que mejoren el desempeño del método en diferentes condiciones del ambiente.
- Se han aplicado estrategias para una correcta detección de objetos, optimizando el método para que funcione en tiempo real.
- El software utilizado permite su portabilidad a diferentes sistemas operativos.
- Se han presentado dos prototipos de aplicación.

Trabajos futuros

- Estudiar soluciones basadas en GPU y múltiples hilos de procesamiento.
- Investigar soluciones para el manejo de la oclusión.
- Explorar el uso de diferentes detectores de características: MSER, FAST, etc.
- Integrar el método en una aplicación con interfaz gráfica para detección sobre un conjunto de imágenes.

Muchas gracias!

Método para detección y seguimiento de objetos con aplicaciones en Realidad Aumentada

Christian Nicolás Pfarher
c.pfarher@gmail.com

sinc(*i*)

Centro de **I**nvestigación en **s**eñales
sistemas e **i**nteligencia **c**omputacional

Ingeniería en Informática - Universidad Nacional del Litoral